

PROCESS CONTROL CONFIGURATION SYSTEM TO BE USED TOGETHER WITH PROFIBUS DEVICE NETWORK

Publication number: JP2001202324 (A)

Publication date: 2001-07-27

Inventor(s): KRIVOSHEIN KENNETH D +

Applicant(s): FISHER ROSEMOUNT SYSTEMS INC +

Classification:

- international: G05B15/02; G05B19/418; G05B23/02; G06F13/00; G06F13/14; G05B15/02; G05B19/418; G05B23/02; G06F13/00; G06F13/14; (IPC1-7): G05B15/02; G05B23/02; G06F13/00; G06F13/14

- European: G05B19/418N

Application number: JP20000305035 20001004

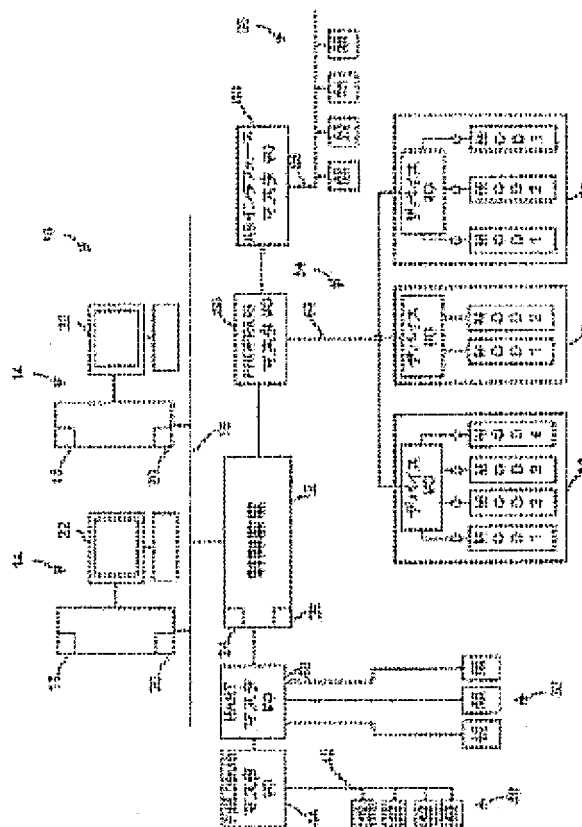
Priority number(s): US19990412037 19991004

Also published as:

GB2358559 (A)
GB2358559 (B)
US6449715 (B1)
HK1062849 (A1)
DE10049049 (A1)

Abstract of JP 2001202324 (A)

PROBLEM TO BE SOLVED: To provide a process control configuration system to be used together with an AS interface device network. **SOLUTION:** Each of AS interface 60, Profibus 55 and HART 48 or Fieldbus 44 has a device network and performs communication with a controller 12 while using a different communication protocol for input/output. An I/O configurator accesses a configuration data base storing configuration information on these networks by an access routine. Further, the block diagram of these device networks is displayed by a document routine.



Data supplied from the *espacenet* database — Worldwide

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2001-202324

(P2001-202324A)

(43)公開日 平成13年7月27日(2001.7.27)

(51)Int.Cl. ⁷	識別記号	F I	テマコード*(参考)
G 0 6 F 13/14	3 3 0	G 0 6 F 13/14	3 3 0 B
G 0 5 B 15/02		G 0 5 B 15/02	A
		23/02	T
G 0 6 F 13/00	3 5 7	G 0 6 F 13/00	3 5 7 A

審査請求 未請求 請求項の数50 O L (全 39 頁)

(21)出願番号 特願2000-305035(P2000-305035)
(22)出願日 平成12年10月4日(2000.10.4)
(31)優先権主張番号 09/412037
(32)優先日 平成11年10月4日(1999.10.4)
(33)優先権主張国 米国 (US)

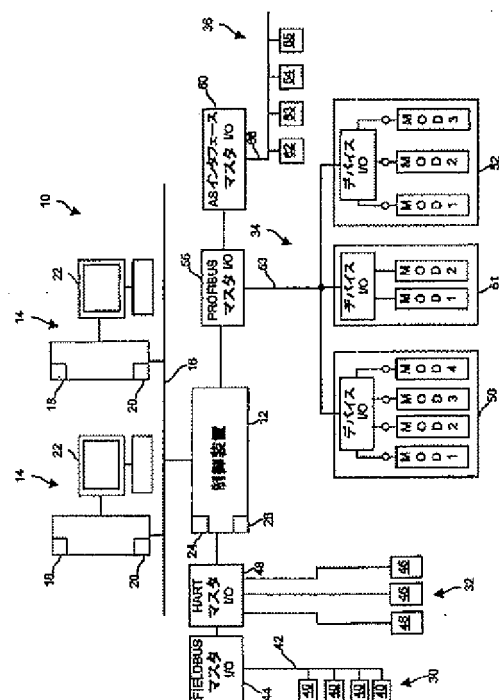
(71)出願人 594120847
フィッシャー・ローズマウント システムズ, インコーポレイテッド
アメリカ合衆国 78754 テキサス オー
スティン キャメロン ロード 8301
(72)発明者 クリヴォシェイン, ケネス ディー.
アメリカ合衆国 78621 テキサス エル
ギン エルギン ウッズ レーン 108
(74)代理人 100065868
弁理士 角田 嘉宏 (外4名)

(54)【発明の名称】 Profibus デバイスネットワークとともに使用するためのプロセス制御構成システム

(57)【要約】 (修正有)

【課題】 ASインタフェースデバイスネットワークとともに使用するためのプロセス制御構成システムを構成する。

【解決手段】 ASインターフェース60、Profibus 55、HART48またはFieldbus 44は、それぞれデバイスネットワークを有し、異なる入出力用通信プロトコルを使用して、制御装置12と通信する。I/Oコンフィグレータは、これらのネットワークに関する構成情報を格納している構成データベースに対してアクセスルーチンによりアクセスする。更に、文書ルーチンによって、これらのデバイスネットワークの構成図を表示する。



【特許請求の範囲】

【請求項1】 制御装置、第1入出力プロトコルを使用して通信する第1デバイスネットワークと、Profibus入出力通信プロトコルを使用して通信する第2デバイスネットワークとを有するプロセス制御ネットワークで使用するための構成システムであって、

構成データベースと、

前記第1デバイスネットワークに関する第1デバイスネットワーク構成情報、およびProfibusデバイスネットワークに関する第2のデバイスネットワーク構成情報を含むデータアクセスルーチンと、

前記Profibusデバイスネットワーク構成情報に基づき前記Profibusデバイスネットワークを構成するコンフィギュレータと、を備え、

前記第1デバイスネットワーク構成情報および前記Profibusデバイスネットワーク構成情報が前記構成データベースに格納される、構成システム。

【請求項2】 前記コンフィギュレータが、前記第1デバイスネットワーク構成情報に基づき前記第1デバイスネットワークをも構成する、請求項1に記載の構成システム。

【請求項3】 前記第1デバイスネットワークがFieldbusデバイスネットワークである、請求項1に記載の構成システム。

【請求項4】 前記第1デバイスネットワークがHARTデバイスネットワークである、請求項1に記載の構成システム。

【請求項5】 前記データアクセスルーチンは、前記Profibusデバイスネットワーク内の第1デバイスに関連するGSDファイルにアクセスして、前記Profibusデバイスネットワーク構成情報の幾つかを入力するファイルアクセスルーチンを含む、請求項1に記載の構成システム。

【請求項6】 前記構成データベースがオブジェクト指向データベースである、請求項1に記載の構成システム。

【請求項7】 前記データアクセスルーチンが前記Profibusデバイスネットワーク用のテンプレートを含み、該テンプレートが前記Profibusデバイスネットワークを構成するために前記Profibusデバイスネットワークのために入手される必要のある前記Profibusデバイスネットワーク構成情報のしるしを格納する、請求項1に記載の構成システム。

【請求項8】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するデバイスのファミリーに関する情報にアクセスする、請求項1に記載の構成システム。

【請求項9】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するデバイスの製造者に関する情報にアクセスする、請求項1に記

載の構成システム。

【請求項10】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するデバイスのモデルに関する情報にアクセスする、請求項1に記載の構成システム。

【請求項11】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するデバイスのモデルのデバイス改訂に関する情報にアクセスする、請求項1に記載の構成システム。

【請求項12】 前記データアクセスルーチンは、前記Profibusデバイスネットワークに関連するデバイスのパラメータに関する情報にアクセスする、請求項1に記載の構成システム。

【請求項13】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するデバイスのモジュールに関連する情報にアクセスする、請求項1に記載の構成システム。

【請求項14】 前記Profibusデバイスネットワークに関連するモジュールのパラメータに関する情報にアクセスする、請求項1に記載の構成システム。

【請求項15】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するモジュールの信号に関する情報にアクセスする、請求項1に記載の構成システム。

【請求項16】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するデバイスのスロットに関する情報にアクセスし、前記スロットは、Profibusデバイスネットワークが存在する位置を示している、請求項1に記載の構成システム。

【請求項17】 前記データアクセスルーチンが、前記Profibusデバイスネットワークに関連するモジュールのデバイス信号タグに関する情報にアクセスする、請求項1に記載の構成システム。

【請求項18】 前記データアクセスルーチンが、前記Profibusデバイスネットワーク内のデバイスに関する情報の階層にアクセスする、請求項1に記載の構成システム。

【請求項19】 前記階層は、デバイス、該デバイスのモジュール、および該デバイスの該モジュールに関連する信号に関する情報を含んでいる、請求項18に記載の構成システム。

【請求項20】 前記階層は、前記信号に対する信号タグを含んでおり、該信号タグは、制御ルーチンを実行するために前記制御装置によって使用され得る、請求項19に記載の構成システム。

【請求項21】 前記Profibusデバイスネットワークは、ProfibusマスタI/Oデバイスを含み、前記コンフィギュレータはProfibusコンフィギュレータであり、該コンフィギュレータは前記Profibusデバイスネットワーク構成情報を使用し

て、前記ProfibusマスタI/Oデバイスを構成する、請求項1に記載の構成システム。

【請求項22】 前記コンフィギュレータが、前記Profibusデバイスネットワーク構成情報内の信号タグによって定義される信号を、前記ProfibusマスタI/Oデバイス内のメモリロケーションにマッピングする、請求項21に記載の構成システム。

【請求項23】 前記コンフィギュレータは、前記Profibusデバイスネットワークを構成するために、前記ProfibusマスタI/Oによって使用されるべきパラメータ化データ文字列と構成データ文字列とを決定する、請求項21に記載の構成システム。

【請求項24】 前記コンフィギュレータは、前記ProfibusマスタI/Oデバイスに自動検知ソフトウェアを格納し、該自動検知ソフトウェアを前記Profibusデバイスネットワーク上に接続されている1つまたは複数のデバイスの存在を自動検出するために使用する、請求項21に記載の構成システム。

【請求項25】 前記構成データベースに格納されている前記構成データに基づいて前記プロセス制御ネットワークの構成概略図を表示する文書化ルーチンを含む、請求項1に記載の構成システム。

【請求項26】 表示される前記構成概略図が、ウィンドウズのエクスポーラタイプの概略図である、請求項25に記載の構成システム。

【請求項27】 前記構成概略図が、前記第1デバイスネットワークの構成を示す第1部分、および前記Profibusデバイスネットワークを示す第2部分を有するシステム構成セクションを含む、請求項25に記載の構成システム。

【請求項28】 前記データアクセスルーチンは前記Profibusデバイスのデバイス定義を作成し、そして、前記コンフィギュレータは前記Profibusデバイスネットワークを構成するために該デバイス定義を使用する、請求項1に記載の構成システム。

【請求項29】 制御装置、第1通信プロトコルを使用する第1デバイスネットワーク、およびProfibus I/Oカードに接続されているProfibusデバイスを有すると共にProfibus通信プロトコルを使用するProfibusデバイスネットワークを含むプロセス制御システムを構成するための方法であって、構成データベース内に格納するための前記Profibusデバイスに関連するデバイス定義を作成し、前記デバイス定義が前記Profibusデバイスに関連する信号に関する情報を含んでいる、ステップと、前記Profibusデバイスのしるしを前記Profibus I/Oカードのポートと関連付けて、前記Profibusデバイスのプロセス制御システムに対する実際の接続を反映する構成文書化システムを使用するステップと、

前記プロセス制御システム内で接続されているように前記Profibusデバイスと関連する前記信号の信号タグを割り当てるステップと、

前記Profibus I/Oカードのポートの構成を前記Profibus I/Oカードにダウンロードするステップと、

制御アプリケーションによって使用される前記信号タグを指定することによって前記信号を使用するように前記制御装置内で実行されるべき制御アプリケーションを構成するステップと、を含む方法。

【請求項30】 デバイス定義を作成する前記ステップが、前記Profibusデバイスに関連するGSDを使用するステップを含む、請求項29に記載の方法。

【請求項31】 デバイス定義を作成する前記ステップが、前記Profibusデバイスに関連する1つまたは複数の入力のリストと、前記Profibusデバイスに関連する1つまたは複数のモジュールのそれぞれに対する1つまたは複数の信号とを構築するステップを含む、請求項29に記載の方法。

【請求項32】 デバイス定義を作成する前記ステップが、前記Profibusデバイスの製造メーカ、モデル、および改訂を特定するステップを含む、請求項29に記載の方法。

【請求項33】 デバイス定義を作成する前記ステップが、前記Profibusデバイスに関連するパラメータの値を指定するステップを含む、請求項29に記載の方法。

【請求項34】 構成文書化システムを使用する前記ステップが、前記Profibusデバイスネットワーク内の前記Profibusデバイスのアドレスを指定するステップを含む、請求項29に記載の方法。

【請求項35】 構成文書化システムを使用する前記ステップが、前記Profibusデバイスに関連する複数のモジュールの順序を指定するステップを含む、請求項29に記載の方法。

【請求項36】 ポートの構成をダウンロードする前記ステップが、前記ポートを介して前記Profibusデバイスと通信するために前記Profibus I/Oカードを構成するステップを含む、請求項29に記載の方法。

【請求項37】 ポートの構成をダウンロードする前記ステップが、前記Profibus I/Oカード内のメモリロケーションに前記信号タグによって定義される信号をマッピングするステップを含む、請求項29に記載の方法。

【請求項38】 ポートの構成をダウンロードする前記ステップが、前記Profibusデバイスを構成するために前記Profibus I/Oデバイスによって使用されるパラメータ化データ文字列と構成データ文字列とを決定するステップを含む、請求項29に記載の方

法。

【請求項39】 前記構成データベースに前記デバイス定義を格納するステップを更に含む、請求項29に記載の方法。

【請求項40】 前記第1デバイスネットワークを示すこととともに、前記プロセス制御システムに対するProfibusデバイスの接続を示すステップを更に含む、請求項29に記載の方法。

【請求項41】 前記オブジェクト指向データベースにおけるオブジェクトとして、前記Profibusデバイスネットワークの構成データおよび前記第1デバイスネットワークの構成データを格納するステップを更に含む、請求項29に記載の方法。

【請求項42】 制御装置、第1入出力プロトコルを使用して通信する第1デバイスネットワーク、およびProfibus入出力通信プロトコルを使用して通信するProfibusデバイスネットワークを有するプロセス制御ネットワークで使用するための構成システムであって、

前記第1デバイスネットワークに関する構成情報、および前記Profibusデバイスネットワークに関する構成情報を格納する構成データベースと、

該構成データベース内での格納のためにProfibusデバイス用にデバイス定義を構築するデータ取得ルーチンであって、前記データ定義が前記Profibusデバイスに関連する信号に関する情報を含む、データ取得ルーチンと、

ユーザが、前記ProfibusデバイスのしるしをProfibus I/Oカードのポートに関連付け、前記Profibusデバイスのプロセス制御ネットワークへの実際の接続を反映することによって所望の構成を示すことができるようにする構成文書化システムと、

前記Profibusデバイスのデバイス定義および示されている所望の構成に基づいて前記Profibusデバイスネットワークを構成するコンフィギュレータとを備えている構成システム。

【請求項43】 前記データ取得ルーチンが、前記Profibusデバイスに関連する前記信号の信号タグを割り当てる、請求項42に記載の構成システム。

【請求項44】 前記コンフィギュレータが、前記Profibusデバイスのデバイス定義に基づき、前記Profibus I/Oカードの構成を前記Profibus I/Oカードにダウンロードする、請求項42に記載の構成システム。

【請求項45】 前記データ取得ルーチンが、前記Profibusデバイスに係るGSDファイルから、前記Profibusデバイスに関するデバイス定義データを取得する、請求項42に記載の構成システム。

【請求項46】 前記データ取得ルーチンが、前記Profibusデバイスに関連する1つまたは複数のモジ

ュールのリストと、前記Profibusデバイスに関連する1つまたは複数のモジュールのそれぞれに対する1つまたは複数の信号とを構築することによってデバイス定義を構築する、請求項42に記載の構成システム。

【請求項47】 前記データ取得ルーチンは、ユーザが前記Profibusデバイスに関連するパラメータの値を指定できるようにする、請求項42に記載の構成システム。

【請求項48】 前記データ取得ルーチンは、ユーザが前記Profibusデバイスネットワーク内の前記Profibusデバイスのアドレスを指定できるようにする、請求項42に記載の構成システム。

【請求項49】 前記コンフィギュレータは、前記Profibus I/Oカード内のメモリロケーションに前記信号をマッピングすることによって前記ポートの構成をダウンロードする、請求項42に記載の構成システム。

【請求項50】 前記コンフィギュレータは、前記Profibusデバイスを構成するために、前記Profibus I/Oデバイスによって使用されるパラメータ化データ文字列および構成データ文字列を決定する、請求項42に記載の構成システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、概してプロセス制御システムに関し、さらに特定すると、構成とともにローカル入出力インタフェースまたは専用(specialized)入出力インタフェースを使用するデバイスネットワークの制御と、ASインタフェースデバイスインタフェースなどの遠隔入出力インタフェースを使用するデバイスネットワークの制御と構成を統合するプロセス制御構成システムに関する。

【0002】

【従来の技術】化学プロセス、石油プロセス、またはそれ以外のプロセスで使用されるシステムのようなプロセス制御システムは、典型的には、アナログおよび/またはデジタルバスまたはその他の通信回線またはチャネルを介して、少なくとも1つのホストまたはオペレータワークステーションに、および1つまたは複数のフィールドデバイスに通信により(communicatively)結合されている少なくとも1つの集中プロセス制御装置を含む。例えば、弁、弁位置決め装置、スイッチ、伝送器(例えば、温度、圧力、および流量センサ)等である場合があるフィールドデバイスは、弁の開閉およびプロセスパラメータの測定などの機能をプロセス内で実行する。プロセス制御装置は、フィールドデバイスによって行われるプロセス測定値を示す信号、および/または入出力(I/O)デバイスを介したフィールドデバイスに関するそれ以外の情報を受信し、この情報を使用し、制御ルーチンを実現してから、プロセスの動作を

制御するために、フィールドデバイスへ入出力装置を介してバスまたはそれ以外の通信路上で送信される制御信号を生成する。

【0003】過去においては、従来のフィールドデバイスは、アナログ回線を介してプロセス制御装置へ、およびプロセス制御装置からアナログ（例えば、4ミリアンプから20ミリアンプ）信号を送受するために使用された。これらの4maから20maの信号は、典型的には、デバイスによって行われる測定、またはデバイスの動作を制御するために必要とされる制御装置によって生成される制御信号を示していた。これらの従来のフィールドデバイスは、典型的には、別個の回線または通信路を介して、代わりに直接制御装置に接続され、制御装置とデバイスの間の通信を使用可能にしたローカル入出力（I/O）デバイスに個々に接続されていた。これらの別個の回線または通信路は、デバイスによって測定された信号を任意の時点で制御装置に送信できるようにするか、あるいは制御装置を任意の時点でデバイスに制御装置によって個別に送信できるようにした。I/Oデバイスが、フィールドデバイスから制御装置へ、または制御装置からフィールドデバイスへ直接送達される信号を多重化するこの構成が、ローカルI/Oと呼ばれている。

【0004】過去10年ほどの間に、プロセス制御業界においては、マイクロプロセッサおよびメモリを含むスマートフィールドデバイスが普及してきた。プロセス内での一次的な機能を実行することに加えて、スマートフィールドデバイスは、デバイスに関するデータを格納し、デジタルフォーマットまたは結合されたデジタルアナログフォーマットで、制御装置および/またはその他のデバイスと通信し、自己校正、識別、診断等の二次的なタスクを実行してよい。HART（登録商標）、PROFIBUS（登録商標）、アクチュエータセンサインタフェース（Actuator Sensor Interface）（これ以降、「ASインタフェース」または「ASI」）、WORLDFIP（登録商標）、Device-Net（登録商標）、CANおよびFOUNDATION（商標）

Fieldbus（これ以降「Fieldbus」）プロトコルなどの数多くの標準的でオープンなスマートデバイス通信プロトコルが、さまざまな製造メーカによって作られるスマートフィールドデバイスを、同じプロセス制御ネットワーク内で一緒に使用できるようにするために開発されてきた。

【0005】一般的に言えば、Fieldbusプロトコルなどのこれらの専用通信プロトコルのいくつかの場合、多数のデバイスがバスまたはネットワークに接続され、バスまたはネットワーク上で（制御装置に接続されている）I/Oデバイスと通信する。Fieldbusプロトコルのケースでは、各デバイスはI/Oデバイスに対し、およびそれによって制御装置に対し1つまたは複数の信号を別個に送信することができる。その結果、

Fieldbusプロトコルは、各デバイスが（個々の信号タグ名等を有する）個々の信号を任意の所望される時点で、あるいは特に指定された時点で通信することができるため、バスを使用し、専用I/Oを実行する。同様に、HARTプロトコルは、各HARTデバイスとI/Oデバイスの間で伸びる別個の回線または通信路を使用し、それによりHART信号を任意の時点でローカルI/Oデバイスに別個に送信できるようにする。その結果、HARTプロトコルはローカルI/O動作を実行する。

【0006】

【発明が解決しようとする課題】しかしながら、ProfibusおよびASインタフェースプロトコルなどのスマートプロトコルの他のものは、一般的にはフィールドデバイスに接続しているI/Oデバイスは制御装置から遠隔に位置し、追加I/Oデバイスを介して制御装置に接続されているため、一般的にはリモートI/Oと呼ばれているものを使用する。要するに、各ProfibusおよびASインタフェースデバイス（またはこれらのデバイスのグループ）は、それに関連しているI/O装置を有する。典型的には、それが関連付けられているデバイス上またはデバイスの近くに位置しているこのI/O装置は、デバイスに関連するさまざまな信号を受信してから、これらの信号を1つの単一データ文字列に連結し、そのデータ文字列を、他のProfibusまたはASインタフェースデバイス、およびしたがって他のProfibusまたはASインタフェースI/O装置が接続されているバス上に載せることによって、これらの信号を多重化する。遠隔I/Oデバイスからのデータ文字列はバス上で送信され、典型的には制御装置の近くに位置しているマスタI/Oデバイスによって受信される。マスタI/Oデバイスはデータ文字列を受信し、これらの文字列をマスタI/Oデバイスに関連するメモリの中に入れる。同様に、マスタI/Oデバイスは、このような信号のセット（つまり、特定のデバイスに送信される信号のすべて）をまとめて連結してから、この連結されたデータ文字列を、遠隔I/Oバス上で、代わりにそれらの信号を復号し、復号された信号を各デバイスの適切なロケーションまたはモジュールに提供するフィールド内のI/O装置に、コマンドおよびその他の信号を遠隔I/Oデバイスのそれぞれに送信する。

【0007】マスタI/Oデバイスは、典型的には、プロセス制御機能を実行する、特別に設計されているプログラム可能論理制御装置（PLC）などの制御装置と接続する。しかしながら、制御装置またはPLCは、特定の信号に関連する個々のデータが、遠隔I/Oフィールドデバイスからデータを受信できるように、マスタI/Oデバイスのメモリ内のどこに格納されているのかを知っていなければならない。同様に、制御装置またはPLCは、遠隔I/Oバス上で遠隔I/Oフィールドデバイ

スに送達されるコマンドおよびその他のデータをマスタ I/O デバイスメモリ内のどこに入れるのかを知っていなければならない。この要件のため、制御装置または PLC 設計者は、どのような種類のデータ（例えば、文字列、浮動点、整数等）が、マスタ I/O デバイス内の各メモリロケーションで格納されるのか、およびマスタ I/O デバイス内の各メモリロケーションにおけるデータが何を表しているのか（例えば、このデータがどの遠隔 I/O フィールドデバイスのどの信号に属しているのか）を追跡調査しなければならない。同様に、制御装置または PLC は遠隔 I/O フィールドデバイスにデータを送信する場合正確な文字列が示された遠隔 I/O フィールドデバイスに確実に送信されるためマスタ I/O デバイス内の適切なメモリロケーション上の適切なタイプのデータを配置するプログラムを作成しなければならない。

【0008】Profibus および AS インタフェースプロトコルなどの大部分の遠隔 I/O 通信プロトコルは、遠隔 I/O バスに置かれるデータ文字列の形式、例えばデータ文字列がどのくらいの長さになるのか、単一データ文字列を形成するためにどのくらいの数の信号を連結できるのか、データ文字列が送信されるボーレート等だけを指定する。しかし、送信されたデータのタイプを指定又は識別しない。このようにして、各 Profibus デバイスの製造メーカは、通常、デバイスに入れることのできるモジュールの数と種類、Profibus バス等上でデバイスに通信されるまたはデバイスから受信される各デバイス信号に関連する入出力データのビットまたはバイトの数などのデバイスについて何らかの情報を有する GSD（ドイツ語の頭字語）ファイルを提供するが、GSD ファイルは、デバイスへ送信およびデバイスから受信されるデータの文字列内のデータが何を表すのかを説明しない。その結果、システムコンフィギュレータは、このデータが表しているのが何の信号なのか、および信号がアナログであるのか、デジタルであるのか、浮動点であるのか、整数値等であるのかを含む、Profibus マスタ I/O デバイスが何を表しているのかを追跡調査しなければならない。同様に、遠隔 I/O バス上で 4 ビットデジタル信号を送信する AS インタフェースデバイスは、デバイスネットワークバス上で送信されているビットのそれぞれが何を表しているのかを知ることまたは理解することをシステム設計者に任せている。

【0009】遠隔 I/O ネットワークによりプロセス制御システムに課される制約のため、制御装置または PLC が、遠隔 I/O フィールドデバイスのそれぞれと関連する信号ごとに選択または確立される（マスタ I/O デバイス内の）メモリロケーションを使用するために構成できるだろうことを保証するために、遠隔 I/O デバイスネットワークを使用する従来の技術によるプロセス制

御システムは、遠隔 I/O デバイスネットワークが、マスタ I/O デバイスとともに、プロセス制御システムの残りとは無関係に構成されることを必要とした。したがって、従来の技術によるシステムで遠隔 I/O フィールドデバイスを使用したプロセス制御システムを構成するためには、システムエンジニアは、まず、所望のフィールドデバイスのすべてと遠隔マスタ I/O デバイスネットワークを遠隔 I/O バスに接続することにより遠隔 I/O デバイスネットワークをセットアップしなければならない。それから、直接的に遠隔マスタ I/O デバイスに接続されているラップトップコンピュータなどのパーソナルコンピュータ上で実行される（例えば、シーメンス（Siemens）によって提供される）使用可能な構成ツールを使用して、構成エンジニアは、遠隔 I/O バスに接続されているデバイスを指定するデータを入力しなければならない。それから、構成ツールは、マスタ I/O デバイスを構成し、それを行う上で、遠隔 I/O フィールドデバイスから受信され、遠隔 I/O フィールドデバイスへ送信されている信号のそれぞれに使用されるマスタ I/O デバイス内のメモリロケーションを選択した。それ以降、いったん遠隔 I/O デバイスネットワークがセットアップされ、マスタ I/O デバイスが構成されたら、エンジニアは、プロセス制御ルーチンまたは機能を実行する一方で、遠隔マスタ I/O デバイス内の適切なメモリロケーションからデータを取得、およびデータを送信するために制御装置または PLC をプログラムしなければならない。言うまでもなく、これは、エンジニアが遠隔 I/O フィールドデバイスのそれぞれに関するデータ（およびマスタ I/O デバイス内のそれらの関連する信号のアドレス）を制御装置または PLC 構成データベースの中に入力することを必要とした。次に、所望される場合、エンジニアは、どの遠隔 I/O フィールドデバイスがシステムに接続されたのか、および制御装置または PLC がマスタ I/O デバイスを介してこれらのデバイスとどのようにして適切に通信したのかに関して文書を提供しなければならない。この複数ステップの構成プロセスは時間を要し、専用 I/O、ローカル I/O または従来の I/O を使用してデバイスと通信するためにプロセス制御システムを構成することとは別個に、それから離れて行われなければならない、少なくとも 2 つおよびおそらく 3 つの別個のシステム内で、2 回または 3 回の異なるときに、つまりマスタ I/O デバイスを構成するとき、マスタ I/O デバイスと適切に通信するために制御装置または PLC を構成するとき、および遠隔 I/O デバイスが通信によって制御装置または PLC に結合された方法を文書化するとき、遠隔 I/O デバイスに関するデータを入力することを必要とした。複数のデータベースに同じまたは類似したデータを入力する要件は、構成または文書化でのエラーにつながるだろう。

【0010】前記に注記されたように、サードパーティベンダは、現在、マスタI/OデバイスがProfibusネットワーク上で通信を提供できるようにするために、必要なデータのあるデータベースを生成することによってProfibusマスタI/Oデバイスを構成するソフトウェアおよび/またはハードウェアシステムを販売している。しかしながら、これらのサードパーティシステムが、マスタI/Oデバイスのどのメモリロケーションにどの信号が格納されているのかに関する文書を提供する程度まで、この文書はProfibusネットワーク内のデバイスに制限され、異なるデータベース内でそのデータを入力し直さずにProfibusプロトコルを使用していないプロセス制御システム内のその他のネットワークによって使用されることはできない。したがって、どの信号がマスタI/Oデバイス内のどのメモリロケーションに格納されるのか、それらの信号が表すのはどのような物理的な現象なのか、およびそれらの信号がどのように構成されているのか（つまり、それらがどのような種類のデータを表しているのか）を追跡調査し、文書化するために必要とされるデータ調整活動は、特に多数のデバイスがProfibus、ASインタフェース、またはそれ以外の遠隔I/Oネットワークに接続されているときに、非常に複雑かつ単調で退屈なものとなる。さらに、適切に文書化されないと、この信号の調整は、遠隔I/Oデバイスネットワーク上でデバイスを構成し直す際に、このような再構成タスクが、必ずやPLCのレジスタのそれぞれの中または制御装置の中の信号のそれぞれが何を表しているのか、およびこれらの信号がマスタI/Oデバイスのメモリからどのようにして入手されるのかを決定し直すことを伴うだろう。制御装置またはPLCの再プログラミングを必要とする可能性があるため、エラーを引き起こすことがある。

【0011】遠隔I/OとローカルI/Oまたは専用I/Oの両方を使用するプロセス制御システムを構成し、文書化するという問題は、さらに、プロセス制御装置およびプロセス制御システムが通常、遠隔I/Oネットワークの通信戦略とは異なる通信戦略を使用して動作するように構成されているという事実によって悪化する。例えば、テキサス州、オースティン（Austin, Texas）に位置するフィッシャーローズマウントシステム社（Fisher-Rosemount Systems, Inc.）によって製造販売されているDeltaV（商標）制御装置システムは、Fieldbusプロトコルに類似した制御通信戦略を使用するように設計されている。特に、DeltaV制御装置システムは、制御交差を実行するために、制御装置または（Fieldbusフィールドデバイスなどの）異なるフィールドデバイスに位置している機能ブロックを使用し、一般的にはデバイス信号タグ（DST）と呼ばれている（典型的には、信号がどこから発したのかを表す）一意の信号タグまたは経路名を与えられた信号を使

用して機能ブロック間の相互接続を指定する。各機能ブロックは、（同じデバイス内、または異なるデバイス内のどちらかの）その他の機能ブロックから入力を受け取るか、および/または出力を提供し、プロセスパラメータの測定や検出、デバイスの制御、あるいは比例導関数積分（proportional-derivative-integral）（PID）制御ルーチンの実現のような制御動作の実行などのなんらかのプロセス制御動作を実行する。プロセス制御システム内の異なる機能ブロックは、1つまたは複数のプロセス制御ループを形成するために互いに（例えば、バス上または制御装置内で）通信するように構成され、その個々の動作はプロセス制御をより非集中型にするためにプロセス全体で広げられてよい。DeltaV制御装置は、Fieldbusデバイスによって使用されている設計プロトコルに非常に類似した設計プロトコルを使用しているため、プロセス制御戦略を、制御装置向けに設計し、その要素を制御装置に接続されているFieldbusデバイスにダウンロードさせることができる。DeltaV制御装置およびFieldbusデバイスは基本的に同じ機能ブロック設計構成子を使用して動作するため、制御装置は、容易にFieldbusデバイスと通信し、Fieldbusデバイス内の機能ブロックからの入信信号を制御装置内の機能ブロックに関連付けることができる。同様に、Fieldbusデバイスおよび専用、ローカル、または従来のI/Oを使用するその他のデバイスは、構成ルーチンが機能ブロック間で送信される信号を指定できるため、共通構成ルーチンを使用して構成することができ、構成されており、そこでは、各信号は、一意の経路またはタグ名を有している。事実上、（専用I/O環境である）FieldbusおよびローカルI/O環境は、デバイスからの各信号の通信を通信路上で制御装置へ別々に可能にするため、制御装置が、信号をこれらのデバイスに送信し、信号をこれらのデバイスから受信し、共通構成データベースを使用するこれらのデバイスを使用してシステムを構成することは、かなり率直である。その結果、DeltaVシステム用の構成ルーチンは、制御装置、Fieldbusフィールドデバイスに関する情報、およびすでにその中に統合されているHARTデバイスなどのそれ以外のローカルのまたは従来のI/Oデバイスに関する制限された情報を有する、すでに結合された構成データベースを提供する。しかしながら、ProfibusプロトコルおよびASインタフェースプロトコルなどの遠隔I/Oデバイス通信プロトコルは、デバイスからの複数の信号に関係するデータの文字列を通信し、制御装置に信号を個別に通信することはできないため、ローカルI/Oデバイスまたは専用I/Oデバイスの制御を提供するように設計されている構成システムの使用は、ローカルI/Oデバイスまたは専用I/Oデバイスに制限され、遠隔I/Oデバイスネットワークには拡張されなかった。

【0012】

【課題を解決するための手段】プロセス制御構成システムは、4-20maなどのローカルI/Oプロトコル、およびHARTプロトコルやFieldbusプロトコルなどの専用プロトコルを使用する制御ネットワークに接続されているデバイスの構成および文書化を、ProfibusおよびASインタフェース通信プロトコルなどの遠隔I/Oプロトコルを使用する制御システムに接続されているデバイスの構成および文書化と統合し、それによって制御システムが、共通構成データベースに基づき異なる通信プロトコルを使用するさまざまな種類のフィールドデバイスと通信し、それを制御できるようにする。特に、プロセス制御構成システムは、ユーザが1つまたは複数の遠隔I/Oデバイスに関するデータを入力できるようにし、遠隔I/Oデバイスのデバイス定義を作成するために、遠隔I/Oネットワークを介してシステムに接続されている遠隔I/Oデバイスのそれぞれに関する情報を入力するようにユーザに自動的にプロンプトを出す。ユーザによって割り当てられる信号タグまたは経路名を含む、遠隔I/Oデバイスのそれぞれに関連する信号に関する情報を含む可能性のある遠隔I/Oデバイス情報は、ローカルI/Oまたは専用I/Oを使用してシステムに接続されているデバイスを含む、プロセス制御システム内のその他のデバイスに関する情報と同じデータベースに格納される。所望される場合、このデータベースは、デバイス、モジュールおよびデバイスに関連する信号を定義するために使用されるオブジェクトの階層を有するオブジェクト指向データベースであってよい。

【0013】(その他のデバイスだけではなく)遠隔I/Oデバイスに関連するデバイス、モジュール、信号等のそれぞれに関する情報を入力した後、構成システムは、プロセス制御システム内の制御装置と遠隔I/Oフィールドデバイス間の通信を可能にする実行時構成を作成し、遠隔I/Oデバイスネットワークに関連するマスタI/Oマスタデバイスにダウンロードする。実行時構成によって、制御装置は、遠隔I/Oフィールドデバイスのそれぞれと関連する信号のそれぞれがマスタI/Oデバイス内のどこに格納されているのか、それらの信号のそれぞれが何を表すのか、これらの信号の性質(つまり、それらがデジタルであるのか、アナログであるのか、浮動点値であるのか、整数値であるのか等)、たとえばこれらの信号が遠隔I/Oバス全体で個別に送信できなくても、制御装置が信号経路または信号タグを遠隔I/Oバス全体で送達される信号のそれぞれに割り当てるために必要とされる情報のすべてを有するように信号に関連する信号名または経路名等を認識できるようにする。

【0014】さらに依然として、構成システムは、それがシステムに接続されているすべてのデバイスに関する

情報を、それらがローカルI/Oデバイスを介して接続されているのか、専用I/Oデバイスを介して接続されているのか、または遠隔I/Oデバイスを介して接続されているのかに関係なく、格納するために同じデータベースを使用するため、遠隔I/Oデバイスの文書化をローカルI/Oデバイスまたは専用I/Oデバイスに自動的に統合する。この文書化は共通構成文書化略図に表示されてよく、ローカルI/Oデバイスネットワーク、専用I/Oデバイスネットワーク、および遠隔I/Oデバイスネットワーク内のデバイスに関する情報を有する。

【0015】本発明の態様に従って、制御装置、(FieldbusまたはHARTデバイスネットワークプロトコルなどの)第1入出力プロトコルを使用して通信する第1デバイスネットワーク、およびProfibus入出力通信プロトコルを使用して通信する第2デバイスネットワークを有するプロセス制御ネットワークで使用するための構成システムは、第1デバイスネットワークに関する構成情報およびProfibusデバイスネットワークに関する構成情報を格納する構成データベースを含む。データアクセスルーチンは、第1デバイスネットワークに関する第1デバイスネットワーク構成情報およびProfibusデバイスネットワークに関する第2デバイスネットワーク構成情報を自動的に要求し、Profibusデバイスネットワークのデバイス定義を作成してよい。それから、コンフィギュレータは、Profibusデバイスネットワーク構成情報に基づきProfibusデバイスネットワークを構成し、Profibusデバイスネットワーク構成情報を構成データベースの中に格納する。

【0016】本発明の別の態様に従って、制御装置、第1通信プロトコルを使用する第1デバイスネットワーク、およびProfibus I/Oカードに接続されているProfibusデバイスを有するProfibusデバイスネットワークを含むプロセス制御システムを構成する方法は、構成データベース内に格納するためにProfibusデバイスに関連するデバイス定義を作成するステップと、Profibusのしるし(indication)をProfibus I/Oカードのポートと関連付け、Profibusデバイスのプロセス制御システムに対する実際の接続を反映するために構成文書化システムを使用するステップを含む。方法は、Profibusデバイスに関連する信号のために信号タグを割り当てるステップと、Profibus I/Oカードのポートの構成をProfibus I/Oカードにダウンロードするステップと、信号タグを使用して制御装置内で実行される制御アプリケーションを構成するステップとを含む。

【0017】

【発明の実施の形態】ここでは図1を参照すると、プロ

セス制御システム10は、イーサネット（登録商標）コネクション等などの通信ネットワーク16を介して（任意の種類のパーソナルコンピュータ、ワークステーション等であってよい）1台または複数台のホストワークステーションまたはコンピュータ14に接続されているプロセス制御装置12を含む。ワークステーション14のそれぞれは、プロセッサ18、メモリ20および表示画面22を含む。同様に、例としてだけ、フィッシャー・ローズマウントシステムズ社（Fisher-Rosemount Systems Inc.）によって販売されているDelta V（商標）であってよい制御装置12は、プロセスの制御を実現するためにプロセッサ24によって使用されるプログラム、制御ルーチン、およびデータを格納するためのプロセッサ24およびメモリ26を含む。制御装置12は、Fieldbusデバイスネットワーク30、HARTデバイスネットワーク32、Profibusデバイスネットワーク34、およびASインタフェースデバイスネットワーク36を含むさまざまなデバイスネットワーク内で多数のフィールドデバイスに結合されている。言うまでもなく、制御装置12は、図1に示されているデバイスネットワークに加えて、あるいはその代わりに、4-20maデバイスネットワークなどの他の種類のフィールドデバイスネットワーク、および他のローカル、専用、または遠隔I/Oデバイスネットワークに接続できるだろう。制御装置12は、そこに格納されている、あるいはそれ以外の場合、そこに関連付けられている1つまたは複数のプロセス制御ルーチンを実現または監督し、デバイスネットワーク30、32、34および36内のデバイスと、およびホストワークステーション14と通信し、プロセスを制御し、プロセスに関する情報をユーザに提供する。

【0018】Fieldbusデバイスネットワーク30は、Fieldbusリンク42を介して、代わりにローカルコネクションを介して制御装置12に接続される（一般的にはリンクマスタデバイスと呼ばれている）FieldbusマスタI/Oデバイス44に接続されているFieldbusデバイス40を含む。一般的には、Fieldbusプロトコルは、フィールドデバイスを相互接続する2線ループまたはバスに標準化された物理的なインタフェースを提供する全デジタル、シリアル、両方向通信プロトコルである。Fieldbusプロトコルは、実際には、これらのフィールドデバイスがプロセス機構全体で分散されているロケーションで（機能ブロックを使用して）プロセス制御機能を実行し、相対的な制御戦略を実現するためにこれらのプロセス制御機能の実行の前後に互いに通信できるようにする、プロセス内のフィールドデバイスにローカルエリアネットワークを提供する。Fieldbusプロトコルは技術で既知であり、とりわけテキサス州、オースティン（Austin, Texas）に本部がある非営利団体であるField

bus財団から出版、配布および入手可能である多数の記事、小冊子、および仕様書で詳細に説明されている。その結果、Fieldbus通信プロトコルの詳細は、ここには詳しく説明されないだろう。

【0019】同様に、HARTデバイスネットワーク32は、標準ローカルバスまたは他の通信回線を介して制御装置12に接続されているHARTマスタI/Oデバイス48に、通信回線を介して接続されている数多くのHARTデバイス46を含む。一般的にはプロセスパラメータを示すアナログ信号およびマスタI/Oデバイスとフィールドデバイス46の間の回線のそれぞれに関する他のデバイス情報を示すデジタル信号を提供するHARTプロトコルも技術で既知であるが、ここにはさらに説明されないだろう。

【0020】Profibusデバイスネットワーク34は、Profibusリンクまたはバス53を介してProfibusマスタI/Oデバイス55に接続されているProfibusスレーブデバイス50、51、および52を含むとして示されている。ProfibusマスタI/Oデバイス55は、標準I/Oインタフェースカードに接続されているProfibusPCMCIAカードの形を取ってよい。一般的には、Profibus-DPプロトコルは、元は、おもにシーメンス（Siemens）によって開発され、後に欧州Fieldbus仕様（European Fieldbus Specification）（EN 50 170）の一部となったドイツ国家規格（DIN 19245）であったプロトコルのファミリーの1つである。このプロトコルのおもな機能とは、モータスタ、電磁弁端末、および可変速度駆動装置などの遠隔I/Oデバイスにインタフェースを提供することである。典型的には、このインタフェースはプログラム可能論理制御装置（PLC）に対してである。Profibus仕様は、デバイス50、51、および52、デバイス55などのDP-マスタ（クラス1）デバイスおよびDP-マスタ（クラス2）デバイス（図1には図示されていない）などのスレーブデバイスを含む3つのクラスのデバイスの動作を記述している。フィールドデバイスは、通常、スレーブデバイスであるが、（制御装置12内の制御アプリケーションのような）制御アプリケーションに対するインタフェースは、例えばマスタデバイス55などのDP-マスタ（クラス1）デバイスを必要とする。DP-マスタ（クラス2）デバイスは、その他のクラスのデバイスの通信能力を構成、診断することができる。しかしながら、ProfibusプロトコルのマスタI/Oデバイスにより実行される構成が、Profibusネットワーク4内のProfibusデバイスの構成に制限され、PLCまたは制御装置12などの制御装置に格納されるあるいは実行される制御アプリケーションの構成、あるいは他のプロトコルに従うフィールドデバイスの構成を含まないことが理解されるだろう。

【0021】関係するプロトコルであるProfibusプロセス自動化(Profibus-PA)は、Profibus-DPに基づき、セグメントカブラを通してProfibus-DPに接続することができる新しい物理層に対するサポート(財団Fieldbus(Foundation Fieldbus)によって使用されているものと同じ)を含む。さらに、Profibus-PAプロトコルは、特にProfibus-PAデバイスのサポートに関して開発されたが、Profibus-DPデバイスにも使用できるProfibus-DPプロトコルの拡張のセットを含む。その結果、図1のProfibusマスタデバイス55は、所望される場合Profibus-PAマスタであってよい。言うまでもなく、現在存在する、あるいは将来開発されるそれ以外の種類のProfibusデバイスおよびプロトコルは、本発明に従って使用されてよい。

【0022】Profibus-DPプロトコルの主要な目的とは、マスタI/Oデバイスとスレーブデバイス50-52のそれぞれの間でデータのセットを周期的に交換することである。一般的には、図1のスレーブデバイス50-52のようなProfibusスレーブデバイスは、きわめて複雑である場合がある。また、スレーブデバイス50-52を使用するフィールドアプリケーションを構成するための標準的な通信機構もない。Profibusネットワーク34内の各スレーブデバイスは、デバイス内のモジュールの数と順序が固定されているコンパクトデバイスであるか、ユーザがデバイス内のモジュールの数と順序を構成してよいモジュラーデバイスのどちらかである。説明のために、図1のスレーブデバイス50と52は、(それぞれそれらに関連付けられている4つの交換可能モジュールと3つの交換可能モジュールを有する)モジュラーデバイスであるが、スレーブデバイス51はそれらに関連付けられている2つの固定モジュールを有するコンパクトデバイスである。

【0023】Profibusリンク53で定期的なデータ交換が発生するようになる前に、スレーブデバイス50-52のそれぞれが構成されなければならない。構成プロセスの間、マスタI/Oデバイス55は、パラメータ化データ文字列の形をした(パラメータ化として知られる)スレーブデバイス50-52のそれぞれにパラメータを送信してから、構成一貫性チェックを実行する。パラメータ化の間、デバイスまたはデバイスのモジュールのそれぞれに関連付けられているパラメータデータがスレーブデバイス50-52に送信される。デバイスパラメータは、まず、モジュール構成の順で、モジュールのパラメータが後に続くメッセージ内に位置している。Profibusデバイスのために情報をサポートすることは、デバイスまたはデバイス内のモジュールに関連するパラメータの記述を含み、列挙されたビットフィールドのために表示にテキストも提供するが、スレー

ブデバイス50-52とマスタI/Oデバイス55間の実際のメッセージはこのような情報は提供せず、それはProfibusリンク53で送信されているデータの意味を識別または理解するためにユーザまたは制御アプリケーションに任されている。

【0024】一貫性チェックの間、各スレーブデバイス50-52用のマスタI/Oデバイス55は構成データのそのコピーを(構成データ文字列として)、マスタI/Oデバイス55からのデータがスレーブデバイス50、51または52内の構成データのコピーに一致することを検証するスレーブデバイスに送信する。一般的に、Profibus-DPデバイス向けの攻勢データは、そのそれぞれが各データ交換メッセージに含まれる入出力バイトの数、およびこれらのバイトが互いに一貫しなければならないかどうか、つまり異なるバイト内のデータが同じときに作成されたのか、あるいは異なるときに作成されたのかを指定する一連の識別子を含む。構成データ内の識別子の順序が、識別子のそれぞれがデータ交換メッセージ中で指定するデータの配置を決定する。モジュラーデバイスの場合、構成データは、特定のデバイスのためにユーザが選択するモジュールの数と順序に基づき生成される。スレーブデバイスの各種の構成識別子は、デバイス製造メーカによって供給されるGSD(ドイツ頭字語)ファイルと呼ばれるデバイスデータベーステキストファイルによって指定される。特にGSDファイルは、名前が指定されたモジュールと各モジュールの識別子のリストを格納し、モジュールの最大数およびデータ交換メッセージ中の入出力バイトの数に関する制限の識別、ポーレートに関する情報、応答回数、プロトコルオプション、診断エラーメッセージコード等も含む。その結果、そのネットワークのマスタI/Oデバイス55の構成を容易にするためには、Profibusネットワーク内のスレーブデバイスごとにGSDファイルを有することが望ましい。

【0025】しかしながら、構成識別子もProfibusデバイスのGSDファイルもマスタデバイスとスレーブデバイス間で交換されるデータの意味論に関する情報を含まない。代わりに、識別子およびGSDファイルはスレーブデバイスに送信され、スレーブデバイスから受信されるデータの長さを指定するだけである。Profibus-DPによって想定されているモデルとは、データがマスタI/Oデバイス55の指定されたメモリロケーションに格納され、このデータにアクセスする制御アプリケーションが意味論およびデータの型を知っているということである。このモデルは、本質的には、レジスタ上で実行されている演算がレジスタ内に格納されているデータの型と一貫していることを確実にすることがユーザまたは制御アプリケーションに任されているPLCレジスタモデルである。

【0026】図1のASインタフェースネットワーク3

6は、ASインタフェースバスまたはリンク66を介して多数のASインタフェースフィールドデバイス62-65に接続されているASインタフェースマスタI/Oデバイス60を含む。一般的には、ASインタフェースプロトコルは、(I/Oモジュールを含む)離散I/Oデバイス62-65をプログラム可能論理制御装置などの制御装置に接続するためにビットレベルセンサバスを使用する。ASインタフェースプロトコルの優れた概要は、アリゾナ州、スコッツデール (Scottsdale, Arizona) のAS-iトレード組織 (AS-i Trade Organization) から入手可能な「アクチュエータセンサインタフェース技術概要」と題されている論文に提供されており、さらにAS-i国際協会 (AS-International Association) がこのバスプロトコル用の仕様を維持し、発行している。ASインタフェース仕様は、バスマスタ (ASインタフェースマスタI/Oデバイス60) とそのホストインタフェースの動作を記述しているので、ここに詳細に説明されないだろう。ただし、ASインタフェースバス66上でセンサおよびアクチュエータ62-65を適切に動作するために、ASインタフェースマスタI/Oデバイス60は、データ交換仕様に加えてアドレスとパラメータ構成仕様を含むASインタフェースマスタ仕様に準拠しなければならない。

【0027】理解されるように、各AS-iインタフェースデバイス62-65には、構成および診断の目的でデバイスを識別するために使用される、ユーザによって指定されるタグ名が割り当てられるだろう。AS-iインタフェースフィールドデバイスが作成され、タグを割り当てられると、構成ルーチンによって選ばれるデバイスの型によってサポートされている有効な入出力ごとに離散I/Oポイントが生成される。ユーザによって変更されてよいデフォルトデバイス信号タグ (DST) もこのようなポイントごとに作成される。I/Oポイントの実行時データは、カレント離散I/Oカードデータに類似して処理される、フィールド値とステータスを含む。I/Oの意味論の意味は入出力以外のシステムによって認識されない (つまり、ステータス/データビットは区別がつかない)。

【0028】ASインタフェースデバイスのデバイス構成は、1から31までであるアドレスの割当て、(I/O構成と識別コードと名前が指定されている2つの4ビット値を有する) デバイス記述、および4つのパラメータビットを含む。ASインタフェースマスタI/Oデバイス60は、デバイスが故障した場合にそのオンライン交換を可能にするため、0というデバイスアドレスがデバイス増設または交換のために確保される。しかしながら、ASインタフェース仕様は、重複するアドレスの検出に備えていないので、同じASインタフェースバス上の異なるデバイスに同じアドレスを使用することを避けるのはユーザの責任である。

【0029】ASインタフェースのI/O構成ビットは、どのビットが有効な入力および/または出力であるのかを指定する。デバイスの識別は、識別コードで補足されている。ただし、デバイスインスタンスもデバイスの型も一意の名称を持たない。したがって、ユーザはデバイスがASインタフェースネットワーク36上の特定のアドレスに構成されている内容と一致しないと判定することはできるが、ユーザは、近接スイッチの一定のブランドまたは型などの特定のデバイス型が特定のアドレスに位置するのを検証することはできない。さらに、ASインタフェースフィールドデバイスは、実際にはそのアプリケーション用のパラメータビットのどれかを使用しない可能性があるが、これらのビットは依然としてデバイスを起動するためにデバイスに書き込まなければならない。ただし、構成中にデバイスに送信されるパラメータビットのどれにも標準的な意味はない。同様に、パラメータビットは、フィールドデバイスから読み取ることとはできないため、ユーザは制御装置またはPLCアプリケーションに対しこれらのビットの値を指定しなければならない。特定のデバイスに関してユーザによって生成されるか、インポートされる定義は、入力、出力およびパラメータビットラベルに加えて、I/O構成と識別コードビットを指定する。

【0030】理解されるように、図1のFieldbusデバイスネットワークおよびHARTデバイスネットワーク30と32は、信号が、これらのデバイスネットワーク内のデバイスのそれぞれからマスタI/Oデバイス44または48に、およびそこから制御装置12へ個別に送信できるという点で、制御装置12とデバイス40と46のそれぞれの間の通信を提供するためにローカルI/Oまたは専用I/Oを使用する。他方、ProfibusネットワークおよびASインタフェースネットワーク34と46は、デバイス信号またはデバイスと関連している信号がバス53と66などの遠隔I/Oバス上でともに多重化されるため、制御装置12と通信するために遠隔I/O活動を使用する。

【0031】言うまでもなく、I/Oカード44、48、55および60は任意の所望されるまたは適切な通信プロトコルまたはデバイスプロトコルでよいが、図1に示されているフィールドデバイスは、センサ、弁、伝送器、位置決め装置等の任意の型のデバイスでよい。さらに、将来開発されるあらゆる規格またはプロトコルを含む、Fieldbusプロトコル、HARTプロトコル、Profibusプロトコル、およびASインタフェースプロトコルの他にも、それ以外の規格またはプロトコルに準拠するフィールドデバイスが図1の制御装置12に結合できるだろう。同様に、複数の制御装置12をシステム10に結合することができ、各制御装置12は1つまたは複数の異なるデバイスネットワークに通信によって結合されてよい。また、Fieldbusデバ

イスネットワークやHARTデバイスネットワークなどのローカルデバイスネットワークまたは専用デバイスネットワークは、ProfibusまたはASインタフェースデバイスネットワークなどの遠隔デバイスネットワークとは異なる制御装置に結合されてよい。

【0032】図1の制御装置12は、一般的に機能ブロックと呼ばれているものを使用して制御戦略を実現するために構成されてよく、そこでは各機能ブロックは、総体的な制御ルーチンの一部（例えば、サブルーチン）であり、プロセス制御システム10内でプロセス制御ルーチンを実現するために通信呼出し済みリンク（communications called links）を介してその他の機能ブロックと一っしょに動作する。機能ブロックは、典型的には、伝送器、センサまたはその他のプロセスパラメータ測定装置と関連する機能などの入力機能、PID、ファジー論理等制御を実行する制御ルーチンに関連する機能などの制御機能、またはプロセス制御システム10内のなんらかの物理的な機能を実行するために弁などのなんらかのデバイスの動作を制御する出力機能の1つを実行する。言うまでもなく、ハイブリッドまたはその他の種類の機能ブロックが存在する。機能ブロックは制御装置12内に格納され、それによって実現されてよく、それは典型的には、これらの機能ブロックが標準的な4-20maデバイス、HARTデバイス、ProfibusデバイスおよびASインタフェースデバイスによって生成される信号のために使用されるかそれらと関連付けられるとき、あるいはFieldbusデバイスの場合に当てはまる可能性のあるフィールドデバイス自体に格納され、それらによって実現されてよいときである。制御システムの説明は、機能ブロック制御戦略を使用してここに提供されるが、制御戦略は、はしご型論理（ladder logic）や任意の標準的なプログラミング言語を含むその他の標準プログラミングパラダイムなどのその他の規約を使用して実現、または設計することもできるだろう。

【0033】前記に注記されたように、過去においては、ユーザは、図1に示されているようなシステム10内で物理的にデバイスを接続した後で、関連するバス上でデバイスと通信するためには、依然としてマスタI/Oデバイス44、48、53および66のそれぞれを構成しなければならず、それからマスタデバイス44、48、55および60と通信し、制御装置12内で制御ルーチンを実行したり、制御装置12内で制御ルーチンに従ってデバイスに出力信号または制御信号を送信するために必要とされる信号を入手するために、制御装置12を構成しなければならなかった。例えば、DeltaVシステムにおいて、ユーザは、制御ルーチンのダウンロード時、あるいはFieldbusマスタI/Oデバイスと関連するポートのダウンロード時に、製造、デバイス型、改訂、デバイス内に格納される機能ブロックなど

のFieldbusデバイスに関する情報をワークステーション14の1つでの構成ルーチン実行で入力できるだろうし、構成ルーチンはFieldbusネットワーク30の動作を可能にするための適切な情報でFieldbusマスタI/Oデバイス44を構成するだろう。HARTマスタI/Oデバイスのチャネル（またはI/Oポート）のそれぞれに関連する信号タグも、構成データベースに格納されていた。制御装置12は、関連するマスタI/Oデバイス内で所望の信号と関連する端末に端に接続するだけで、あるいはFieldbus I/OデバイスのケースではFieldbusネットワーク30と制御装置12の間で一貫していたタグ単位で機能ブロックにアクセスすることによって、HARTまたは従来の4-20ma I/Oデバイス上の信号にアクセスできるだろう。構成データは、例えば、ワークステーション14の1つ内にあり、標準的な方法でユーザによってアクセスされる構成データベースに格納されていた。

【0034】しかしながら、遠隔I/Oデバイスネットワークの場合、ユーザは、（マスタデバイスに接続している標準ツールを使用して）マスタI/Oデバイスを手動で構成しなければならず、それから特定のデバイスに関連する特定の信号が、マスタI/Oメモリ内のどこに格納されているのか、およびそれらの信号が何を表しているのかを制御装置12に知らせるために、マスタI/Oデバイスと通信することができるようになるように制御装置12をプログラムしなければならないだろう。このプロセスは、マスタI/Oデバイスの構成が変更されるたびに繰り返されなければならず、多くのエラーにつながり、ProfibusデバイスネットワークとASデバイスネットワーク34と36内でのデバイスの追加または変更を時間を要し、単調で退屈なものにした。同様に、ユーザは、ユーザがこれらのネットワークの構成を見ることができるようになるには、構成データベース内のProfibusデバイスネットワークとASインタフェースネットワーク34と36に関する適切な情報のすべてを入力し直さなければならなかった。ただし、このデータベースはこれらのネットワークの構成を変更するために使用することはできず、例えば、そもそもデータを入力する際にエラーが起きた場合には正確でさえなかった可能性がある。そのすべてが本発明の譲受人に譲渡され、そのすべてがここに参照して明示的に組み込まれているDoveらに対する米国特許第5,838,563号（「プロセス制御環境を構成するためのシステム（System for Configuring a Process Control Environment）」）、Nixonらに対する米国特許第5,828,851号（「標準デバイスおよび標準外デバイスの標準プロトコル制御を使用するプロセス制御システム（Process Control System Using Standard Protocol Control of Standard Devices and Nonstandard Devices）」）、1

996年4月14日に提出されたNixonらに対する米国出願番号第08/631, 519号(「デバイスのネットワークへの接続を自動的に検出するための方法および装置を含むプロセス制御システム(Process Control System Including a Method and Apparatus for Automatically Sensing the Connection of Devices To a Network)」)および1996年4月12日に提出されたDoveに対する米国特許出願番号第08/631, 458号(「プロセス制御環境構成を補助するためのシステム(System for Assisting Configuring a Process Control Environment)」)は、プロセス制御システム内でのデバイスの構成、自動検出および制御を、ローカルI/Oデバイスネットワークまたは専用I/Oデバイスネットワークを使用して実行できる方法を説明する。

【0035】図2は、プロセス制御システム10内のデバイスのすべて用の構成情報を格納する構成データベース72を使用するプロセス制御構成システムまたはルーチン70を示す。構成システム70は、例えば、ホストデバイス14の内の1つのメモリ20の内の1つまたは複数に格納され、ネットワーク30と32などのローカルI/Oデバイスネットワークまたは専用I/Oデバイスネットワークとともに、ネットワーク34と36などの遠隔I/Oデバイスネットワークを構成し、文書化するために、ホストデバイス14のプロセッサ18上で実行されてよい。構成データベース72は、ワークステーション14のメモリ20の内の1つで、またはバスに接続されているスタンドアロンメモリ内のように、任意の所望メモリ内に位置してよい。ただし、構成データベース72は、構成システム70によってアクセス可能でなければならない。構成システム70は、Profibusデバイスネットワーク34とASインタフェースデバイスネットワーク36などの遠隔I/Oデバイスネットワークの構成および文書化を、Fieldbusデバイスネットワーク30とHARTデバイスネットワーク32などの従来のI/Oデバイスネットワークの構成および文書化と調整するように、図1に示されているプロセス制御システム10を構成するために、構成データベース72とともに使用することができる。

【0036】構成システム70は、図1に示されているプロセス制御システムの構成および文書化を実行するためにともに動作するソフトウェアルーチンなどの複数の構成要素を含む。一般的には、構成システム70は、プロセス制御システム10内で、それらのデバイスがプロセス制御システム10内で接続されるように、デバイス(およびそれらのデバイスに関連するモジュール、信号、パラメータ等)のどれか、またはすべてに関する情報を入力するようにユーザにプロンプトを出す、あるいはそれ以外の場合ユーザが入力できるようにするユーザ入力セクション(またはデータアクセスまたは取得セクション)74を含む。構成システム70は、図1のデバ

イス44, 48, 55および60などの異なるマスタI/Oデバイスを構成するコンフィギュレータ76、および構成データベース72に格納されるように、現在の構成に関する文書をユーザに表示する文書化ルーチン78も含む。文書化ルーチン78も、ユーザが後述されるようにプロセス制御システム10の構成を操作、変更できるようにし、ユーザがデバイスの追加、デバイスの削除、デバイス構成の変更等を行うことができるようにするためにユーザ入力ルーチン74とともに使用されてよい。

【0037】一般的に、ユーザ入力ルーチン70は、例えば、デバイスがシステム10に追加される、デバイスがシステム10内で移動される、あるいはなんらかの方法で変更されるたびに必ず、プロセス制御システム10内の任意の要素に関する構成データを入力するように呼び出されることがある。デバイスの自動検出がプロセス制御システムでサポートされている場合、ユーザ入力ルーチン74は、ユーザに、ネットワーク10に接続されていると検出されたデバイスに関する画面または質問を自動的に提示してよい。所望される場合、ユーザ入力ルーチン74は、デバイスを追加する、またはデバイスを変更するなどによってプロセス制御システム10の構成に変更を加えるために文書化ルーチン78が使用されるたびに、呼び出されてもよい。呼び出されると、ユーザ入力ルーチン74は、プロセス制御システム10の実行時にフィールドデバイスと制御装置またはその他のデバイスの間の通信を確立する、または可能にするために、および構成を文書化するために、デバイスまたはデバイスネットワークを構成するために必要となる情報を入力するように自動的にユーザにプロンプトを出す。所望される場合、入力ルーチン74は、遠隔I/Oネットワーク内のさまざまなデバイスのそれぞれにデバイス定義を作成または更新してよく、そこではこのデバイス定義がデバイスを文書化および/または構成するために必要とされるデータを格納する。

【0038】デバイスネットワーク内でのさまざまなデバイスのそれぞれに関する正しく、必要な情報を取得するため、ユーザ入力ルーチン74は、デバイスまたはネットワーク情報を入力または変更するためにユーザ入力ルーチン74によって使用される質問またはその他のダイアログを所望される方法で格納する異なるネットワークテンプレート80-86を使用してよい。情報が、図1のネットワーク30, 32, 34および36などの異なるデバイスネットワークのそれぞれの中のデバイスを構成、文書化することを必要としたため、テンプレート80-86のそれぞれはそのプロトコルに関して必要とされる、またはそれに関連する異なる型のデータを取得するために使用されるさまざまな情報を格納してよい。どのような場合においても、ユーザ入力ルーチン74は、これらの異なる種類のデバイスネットワークおよび

これらのネットワーク内のデバイスのそれぞれを構成、文書化するために必要な特定の情報を要求するために、テンプレート80-86に格納されるデータを使用する。Profibusテンプレート80、ASインタフェーステンプレート82、Fieldbusテンプレート84、およびHARTテンプレート86は図2に示されているが、それ以外のテンプレートまたはインタフェース制御はその他のデバイスネットワークに使用することができるだろう。所望される場合、テンプレート80-86のそれぞれは、関連するデバイスネットワーク内の異なる種類のデバイスのそれぞれに必要とされるその他の情報、そのネットワークまたはそのネットワーク内のデバイスを構成するために必要な情報、および制御装置12がそのネットワーク内のデバイスと効果的に通信できるようにする情報に関する画面表示、質問またはその他のデータを格納してよい。他の所望のダイアログがプロセス制御ネットワーク内のデバイスに関してユーザから情報を入手するために使用されてよいが、図7から図26は、ここに、ProfibusおよびASインタフェーステンプレート80と82を使用して生成されるか、それらの中に格納されてよい例の画面表示を提供する。

【0039】このようにして、理解されるように、ユーザ入力ルーチン74は、デバイスをシステム10に接続する方法、それらのネットワークを構成するために必要なデバイスの型およびその他の情報を含む、ワークステーション14の内の1つを介してデバイスネットワークのそれぞれの中のあるさまざまなデバイスのそれぞれを構成し、それぞれと通信するために必要とされる情報のすべてを入力するように、ユーザにプロンプトを出す。Profibusデバイスネットワーク34の場合でのようないくつかのケースでは、入力ルーチン74は、ユーザが、GSDファイルまたは（デバイス記述などの）その他のデバイス製造メーカファイルを構成システム70に提供し、GSDファイルからそのデバイスのために情報を入手することができるようにしてよい。GSDまたはその他の製造メーカのファイルは、製造メーカファイル記憶装置88に格納されてよいか、あるいは代わりに構成データベース72または任意のそれ以外の所望されるロケーションに格納されてよい。デバイスに関してGSDファイルがすでに存在している場合、あるいはこのようなファイルが構成システム70に提供された後、入力ルーチン74は、Profibusネットワークを構成するために必要とされるデータのいくつか、つまりProfibus構成テンプレート80に格納されるテンプレートを書き入れる（fill out）ために必要とされるデータのいくつかにデフォルト値を書き入れるまたは提供するためにGSDファイル内の情報を使用してよい。言うまでもなく、他の製造メーカのファイルは他の種類のI/Oネットワークおよびデバイスのために存在してよ

く、これらのファイルは、構成ルーチン70にデバイス情報を提供するタスクをさらに簡略にするために使用することができる。

【0040】特定のデバイスに必要な情報を取得した後、ユーザ入力ルーチン70は、例えば、オブジェクト型構造でプロセス制御システム10内のデバイスのそれぞれに関する情報を格納するオブジェクト指向データベースであってよい構成データベース72に受け取られた情報を格納する。オブジェクト指向データベース72のオブジェクトフォーマットは任意の所望されるフォーマットでよいが、このオブジェクトフォーマットは、通常、デバイスの論理配列および各デバイスネットワークに関連付けられているデバイス内の装置に基づかなければならない。言うまでもなく、オブジェクトフォーマットは、システム10に接続されている異なる種類のデバイスネットワークのそれぞれに関して異なってよい。このようにして、オブジェクトは、各デバイスネットワーク内のデバイスごとに作成されてよく、デバイスモジュールに関するサブオブジェクト、機能ブロック、それらのデバイスに関連する信号等は、このようなデバイスオブジェクトごとに提供されてよい。典型的には、ユーザ入力ルーチン74および特定のデバイスプロトコルのテンプレートは、構成データベース72のオブジェクト指向フレームワーク内の各オブジェクトに関連する情報などの、各デバイスの構成データベース72に格納される情報を取得するために構成されるだろう。例えば、図3に示されるように、Profibusデバイスのオブジェクト構造は、Profibusデバイスのファミリーを識別するファミリーオブジェクト、デバイス製造メーカを識別する製造メーカサブオブジェクト、特定のデバイス製造メーカのデバイスのモデルを識別するモデルサブオブジェクト、およびデバイスモデルに関連するデバイス改訂を識別する改訂サブオブジェクトを含む。各デバイス改訂は、デバイスと関連するパラメータを定義する、1つまたは複数のデバイス全体でのパラメータサブオブジェクトを有してよい。同様に、各デバイス改訂は、それと関連する1つまたは複数のモジュールサブオブジェクトを有してよい。同様に、各モジュールは、1つまたは複数のモジュールパラメータサブオブジェクトを有してよく、各モジュールパラメータはそれに関連する1つまたは複数の信号サブオブジェクトを有してよい。各種オブジェクトのための唯一のボックスだけが図3に示されているが、各ファミリーは複数の製造メーカを有してよく、各製造メーカは複数のモデルを有してよく、各モデルは複数のデバイス改訂等を有してよい。

【0041】同様に、図1のネットワーク36などのASインタフェースネットワークは、例えば、各ASインタフェースデバイス型のオブジェクトを含むオブジェクト構造、および任意のデバイス型のデバイスに関するサブオブジェクト、およびそれらのデバイスのそれぞれ

に関連する（離散I/O信号などの）信号とパラメータを使用して編成されてよい。言うまでもなく、各オブジェクトは、そのオブジェクトに関する情報を含む、あるいは格納してよい。例えば、デバイスオブジェクトは、そのデバイスの構成とパラメータ化データ文字列、デバイスの記述、製造メーカ情報、信号タグなどのユーザによって割り当てられているタグ等の構成およびパラメータ化の情報を含んでよい。同様に、モジュールおよび信号オブジェクトは、記述、タグ、およびそれらの装置に関するその他の情報を含んでよい。ProfibusおよびASインタフェースネットワークオブジェクトのために取得、格納されてよい特定の情報のいくつかは、図7から図26に関してこれ以降さらに詳細に説明される。同様に、デバイスネットワークのオブジェクトは、本発明に従ってそれ以外の所望される階層で編成されてよい。言うまでもなく、構成データベース72は、Fieldbusデバイス、HARTデバイス、4-20デバイス、およびシステム10内のそれ以外のデバイスに関するオブジェクトも含んでよく、これらのオブジェクトはDeltaVシステムなどのプロセス制御システム内で現在使用されているオブジェクトデータベースに使用されているものと同じまたは類似してよい。例えば、Fieldbusデバイスは、製造メーカ、デバイス型、改訂、機能ブロック、通信関係、実行時間、機能ブロックのインデックス、機能ブロックの数に関する構成情報、あるいは各Fieldbusデバイスに関連するその他の情報を有してよく、この情報は構成データベース72内のFieldbusデバイスごとにデバイス定義として定義されてよい。HARTデバイスは、例えば、製造メーカ、デバイスタイプ、改訂、記述、デフォルト変数、デバイス識別情報、診断コマンド、デフォルト値、またはHARTデバイスに関連するその他の情報に関する構成情報を有してよく、この情報は構成データベース72内のデバイスごとのデバイス定義として格納されてよい。

【0042】再び図2を参照すると、特定のデバイスネットワーク内のデバイスの1つまたは複数に関する情報をユーザが入力した後、ユーザが制御装置への制御スキームのダウンロードを希望するとき、ユーザがデバイスネットワーク上のデバイスに対する通信リンクの確立を希望するとき、あるいはそれ以外の任意の所望の時点で実現されてよいコンフィギュレータ76は、デバイスネットワークを構成するため、およびそれによって制御装置12とデバイスネットワーク内の1つまたは複数のデバイスの間の通信を可能にするために実現される。一般的には、コンフィギュレータ76は、構成データベース72に格納されている情報を使用して、Profibusネットワーク34のマスタI/Oデバイス55またはASインタフェースネットワーク36のマスタI/Oデバイス60などの特定のデバイスネットワークと関連す

るI/Oデバイスを構成するために使用される。コンフィギュレータ76は、構成される異なる種類のデバイスネットワークのそれぞれに異なる構成ルーチンを格納し、使用してよい。例えば、図2は、Fieldbus、HART、Profibus、およびASインタフェースデバイスネットワークのそれぞれに異なる構成ルーチンを有するコンフィギュレータ76を示す。言うまでもなく、これらの異なるデバイスネットワークの所望される構成ルーチンは使用されてよいが、この構成ルーチンが、ユーザ入力74を介してデバイスネットワークに関してユーザによって入力される、および/または構成データベース72に格納されるような情報を使用することが理解されている。制御装置12とProfibusデバイス50-52の間の通信を可能にするために、例えば、ProfibusマスタI/Oデバイス55を構成した後、コンフィギュレータ76は、ASインタフェースI/Oデバイス60を構成するために1つの異なる構成ルーチン、および必要な場合には、これらのネットワーク内のデバイスについて構成データベース72に格納されている情報を使用してFieldbusマスタI/Oデバイス44およびHARTマスタI/Oデバイス48を構成するために1つまたは複数の依然として異なる構成ルーチンを使用してよい。言うまでもなく、Profibus I/Oデバイス55、ASインタフェースI/Oデバイス60等を構成するために使用される構成ルーチンは、この情報を使用して、既知の方法または所望される方法で適切なI/Oデバイスを構成する制御装置12に構成情報が伝えられなければならない旨を理解して、独自にこれらのデバイスを構成するために現在使用されている構成ルーチンに類似するまたは同じである場合がある。

【0043】このようにして、例えばコンフィギュレータ76は、例えばProfibusデバイスネットワーク36に関連する構成ルーチン、およびユーザによって入力され、Profibusデバイス50、51、および52のそれぞれにデータを送信し、それぞれからデータを受信するために使用されるI/Oデバイス55内のメモリロケーションを選択するためにProfibusネットワーク34内で接続されているProfibusデバイス50-52のそれぞれに関する構成データベース内に格納されている情報を使用してよい。同様に、コンフィギュレータ76は、Profibusデバイスネットワーク34が動作できるようにするために、ProfibusマスタI/Oデバイス55のメモリ内で、Profibusネットワーク34の中のデバイス50-52のそれぞれを構成するために必要とされるパラメータ化データおよび構成データをアSEMBルし、格納してよい。データは、所望される場合、制御装置12内に格納されてもよい。ProfibusマスタI/Oデバイス55が構成される方法、つまり各デバイスの信号がP

Profibus マスタ I/O デバイスメモリ内のどこに格納されるのかに関するメモリ情報も、構成データベース 72 の中に格納され、マスタ I/O デバイス 55 と通信し、実行時中にデバイスとの通信を達成するために使用される制御装置 12 によってアクセスできるようにされてよい。所望される場合、このメモリ情報は、Profibus マスタ I/O デバイス 55 が構成されるとき、または特定の信号を使用する制御ルーチンが Profibus マスタ I/O デバイス 55 から読み取られる、またはそれに読み込まなければならない特定の信号を使用する制御ルーチンが制御装置 12 にダウンロードされるときに、制御装置 12 に提供されてよい。このようにして、ユーザは、Profibus デバイス 50-52 に関する情報を一度だけ入力し、このデータは構成データベース 72 の中に格納され、制御装置 12 が Profibus マスタ I/O デバイスを介してそのネットワーク内でデバイスと通信し、プロセス制御システム 10 の構成を文書化できるように Profibus デバイスネットワーク 34 を構成するために使用される。

【0044】Profibus マスタ I/O メモリに格納されている信号のそれぞれに関連する信号タグ等に関する構成情報は、制御装置 12 がマスタ I/O デバイス 55 内の正しいメモリロケーションにアクセスし、信号、モジュールタグ、デバイスタグ、または名前をそのデータ（このようなタグは構成データベース 72 によって指定されている）に、そのデータが制御装置 12 によって任意の便利な方法で使用できるように割り当てることができるように、自動的に制御装置 12 に提供される。言い替えると、制御装置 12 は、Profibus マスタ I/O デバイス 55 のメモリロケーションのそれぞれに格納されているデータを解釈し、そのデータを、それが Fieldbus デバイスネットワーク 30 や HART デバイスネットワーク 32 などのそれ以外の従来の I/O デバイスネットワークから受け取られるデータまたは信号を使用するのと同じように使用するのに十分な情報を提供される。同様に、制御装置 12 は、そのデバイスネットワークに関連する I/O マスタデバイスの適切なメモリロケーションに送信されるデータを配置することによって、Profibus デバイスネットワーク 34 内の特定のデバイスまたはモジュールにデータまたは信号を送信することができる。所望される場合、Profibus またはその他のデバイス I/O ネットワークに関連するこれらのメモリロケーションは、構成データベース 72 に格納され、制御ルーチンが制御装置 12 にダウンロードされるときに、制御装置 12 がこれらのデバイスとどのように通信しなければならないのかを指定するために使用されてよい。

【0045】図 4 を参照すると、例えば、Profibus ネットワーク 34 用のマスタ I/O デバイス 55 内のメモリであってよいメモリ 90 が示されている。この

ケースでは、コンフィギュレータ 76 は、Profibus デバイスネットワーク 34 内で接続されるデバイスのそれぞれに送信され、それぞれから受信される信号のそれぞれを格納するための特定のメモリロケーションを割り当てる。これらのメモリロケーションは、Profibus デバイスネットワーク 34 に接続されているデバイス 50、51 等であるデバイス 1、デバイス 2 のそれぞれに対するインデータまたはアウトデータとラベルがつけられていると示されている。言うまでもなく、コンフィギュレータ 76 は、言うまでもなくデバイスの型、デバイスに関連するモジュールの数、各モジュールに関連する信号の数と種類等に依存するデバイスから受信されるデータ（インデータ）の連結文字列内のデータのすべてだけではなく、デバイスに送信されるデータ

（アウトデータ）の連結文字列内のすべてのデータに十分な記憶領域があることを保証するためにこれらのメモリロケーションを選択する。しかしながら、この情報のすべては、ユーザ入力セクション 74 によって要求を介して直接的に、またはデバイスに関連する製造メカファイル 88 内の情報に基づき入力され、構成データベース 72 の中に格納される。同様に、Profibus デバイスネットワーク 34 がコンフィギュレータ 76 によって決定されるときにデバイスのそれぞれをパラメータ化し、構成するために必要とされるパラメータ化および構成データは、Profibus デバイスネットワーク 34 内でさまざまなデバイスのそれぞれとの通信を確立するためにマスタ I/O デバイス 55 によって使用される特定のメモリロケーションでメモリ 90 内に配置される。メモリマップ 92 は、制御装置 12 が、マスタ I/O デバイス 55 のメモリ 90 内で信号のそれぞれを解釈できるようにするために制御装置 12 内に格納されてよく、このマップ 92 は、プロセス制御ルーチンを実現するときには制御装置 12 によって必要とされてよい信号タグ情報等を含んでよい。同様に、メモリマップ 92 は、制御装置 12 にメモリ 90 内のデータ文字列情報を復号し、Profibus ネットワーク 34 に接続されている Profibus フィールドデバイス 50-52 の 1 つに送信される適切なデータ文字列フォーマットでメモリ 90 内に情報を配置するために必要な情報を提供してよい。言うまでもなく、詳細は、AS インタフェースプロトコルは、異なる種類の構成データ文字列だけではなく 4 ビット信号文字列も使用するという事実のために異なるが、類似する種類のメモリマッピングは、AS インタフェースマスタ I/O デバイス 60 で実行されてよい。

【0046】言うまでもなく、コンフィギュレータ 76 は、遠隔 I/O マスタデバイスを構成し、遠隔ネットワークに関連するオブジェクト、デバイスまたは信号に関する必要な情報を制御装置 12 に、およびその逆に伝達するために必要な機能を実行する。したがって、コンフ

ィギュレータ76はデバイスに関連する情報またはリモートネットワーク内の信号を、制御装置12内で使用される信号にマッピングするためにリモートI/Oデバイス用のメモリマップを確立してよい。コンフィギュレータ76は、例えばデータ中の変更が検出されると、デバイスから受け取られるデータを制御装置12へ自動的に送信するために、マスタI/Oデバイス内の通信オブジェクトを確立またはセットアップしてもよい。マスタI/Oデバイスの構成の詳細は、異なるプロトコルには異なるが、技術で周知であるため、ここにはさらに説明されないが、この構成が、システムの残りが構成されるとき、またはユーザが遠隔ネットワークに関する新しい情報を入力するとき、あるいはそれ以外の任意の適切なまたは所望される時間にコンフィギュレータ76によって自動的に実行されることが理解されている。

【0047】言うまでもなく、類似する構成活動は、必要な程度まで、図1のASインタフェースデバイス62-65に関して入力されるデータを使用してASインタフェースマスタI/Oカード60のために実行できる。実際、ASインタフェースプロトコルはさらに単純であるため、コンフィギュレータ76は、実行時中に制御装置12によって使用されるために、ASインタフェースマスタI/Oデバイス60の中に、またはその中からデータをマップし、例えば、制御装置12内または構成データベース72内のこれらのメモリロケーションのしるしを格納するだけでよい。コンフィギュレータ76は、例えばデバイスのために作成されたデバイス定義から、ASインタフェースデバイスごとのデバイスプロファイル(つまり、I/O構成コードおよび識別コード)を決定し、この情報をASインタフェースマスタI/Oデバイスに提供してもよい。ASインタフェースマスタI/Oデバイスは、それからその構成コードに関して各ASインタフェースデバイスをポーリングし、そのコードをコンフィギュレータ76によって提供されたコードに比較し、それらが一致する場合には、デバイスパラメータをASインタフェースデバイスに送信し、そのデバイスと通信を開始する。

【0048】再び図2を使用すると、文書化ルーチン78が構成データベース72に格納されているデータに基づいてプロセス制御システム10の現在の状態を表示するために使用できるか、あるいは構成を指定するため、またはプロセス制御システム10の構成を変更するために、ユーザ入力ルーチン74と一しょに使用されてよい。文書化ルーチン78は、構成データベース72内に格納されているデバイスおよびその他のネットワークの情報を、DeltaV制御システムによって現在実行されているような、ウィンドウズ-エクスプローラ型ツリーセットアップを使用するなどの所望の方法で表示することができる。ただし、文書化およびプロセス制御ネットワーク10のセットアップに関する文書の表示のそれ

以外の方法も使用することができる。

【0049】文書化ルーチン78によって作成されてよい、プロセス制御システム10内で接続されているデバイスとその他の要素の階層ビューを示す文書化ツリーまたは構造の例は、図5および図6に示されている。図5と図6の階層ネットワークは、ローカルI/Oデバイスネットワークまたは専用I/Oデバイスネットワークだけではなく、リモートI/Oデバイスネットワークを含む。図5および図6に示されている階層構造は、過去に構成されたか、作成されたことのあるFieldbusデバイス、HARTデバイス、ProfibusデバイスおよびASインタフェースデバイスなどのさまざまなデバイス構成、デバイス定義またはそれに関連するオブジェクトが格納されている構成データベース内に典型的に存在するライブラリを含む。例えば、DeltaV階層の中にすでに提供されているように、Fieldbusデバイスは、製造メーカ、デバイス型、デバイス改訂、デバイス内の機能ブロック、機能ブロックの名前、実行時間、およびデバイス内の機能ブロックのインデックスを有するとして分類されていると示されている。同様に、HARTデバイスは、図5の中で、製造メーカおよびデバイス型に従って分類されていると示されている。各デバイス型は1つまたは複数の識別子を有し、各デバイスは、記述およびそれに関連する特種診断コマンドを有することがある。言うまでもなく、他の構成情報は、ライブラリ内のFieldbusまたはHARTデバイス(またはその他の型のデバイス)について提供されてよい。本発明に従って、(図1のデバイス50-52であってよい)Profibusデバイスは、デバイスのファミリー(図5に示されているFAM1だけ)に該当するとして分類されてよく、デバイスの各ファミリーは1つまたは複数の製造メーカ(MAN1だけが図5に示されている)を含んでよい。モデル(MODEL1だけが示されている)は、さらにProfibusデバイスを分類するために製造メーカに関連付けられてよい。同様に、デバイスの各モデルは、1つまたは複数のデバイス改訂(REV1が示されている)を有し、各デバイス改訂はそれに関連する1つまたは複数のデバイス全体でのパラメータも有してよい(PARAM1が示されている)。デバイス全体のパラメータは、他のデバイス内の制御装置システムによってすでに認識されているものとは異なる方であるProfibusデバイスに関連するパラメータを定義するために使用することができる。同様に、各デバイス改訂は、それと関連する1つまたは複数のモジュールを有してよい(MODULE1が示されている)。モジュールは、Profibusデバイス内にある特定の種類のカードを指す。さらに、各モジュールは、再び、Profibusモジュールに関連する新しい種類のパラメータを定義する1つまたは複数のモジュールパラメータ(PARAM2が示されている)を

有してよく、それに関連するゼロまたは複数の信号を有してよい。これらの信号は、デバイスまたはデバイスのモジュールへの実際の入力あるいはデバイスまたはデバイスのモジュールからの実際の出力である。理解されるように、Profibusデバイスに関する図5および図6の文書階層は、図3のものなどのProfibusデバイスに定義されているオブジェクト構造に従うか、あるいはその元で編成されてよい。

【0050】同様に、図5のライブラリは、図5に図示されているように、システム内で接続されているASインタフェースデバイスの1つまたは複数のしるしを含むことがある。ASインタフェースデバイスは、ASIデバイス型に従って分類されてよい(Device Type 1だけが図5に示されている)。言うまでもなく、ASインタフェースデバイスに関連する製造メーカ、デバイス改訂等の他の分類も提供できるだろう。

【0051】図6の階層は、デバイスが、制御システム10内でどのように物理的に接続されるのかを示すシステム構成セクションを含む。例えば、システム構成は、その下で制御ネットワークセクションが、さまざまなデバイスとデバイスネットワークを制御するために制御装置がどのようにセットアップされているのかを示す物理ネットワークセクションを含んでよい。制御ネットワークセクションの元では、1つまたは複数の制御装置を一覧表示してよい。制御装置は、そこに制御ルーチン(図示されていないが、通常割り当てられたモジュールと呼ばれている)を含み、各制御装置と関連して、I/Oセクションは、制御装置がI/O活動を実行するために通信する、制御装置に接続されているデバイスを定義する。それぞれの異なる種類のデバイスネットワークは、専用のI/Oエントリを有してよい。図1のFieldbusマスタI/Oカード44に対応するFieldbus I/Oカードは、そのポートP01に接続されているD01-D04とラベルが付けられたFieldbus デバイスを有し、これらのデバイスが図1のデバイス40に対応する。これらのデバイスのそれぞれは、それに関連する機能ブロックを有してよい。図1のHARTマスタI/Oカード48に対応するHART I/Oカードは、それぞれチャネル(通常、ワイヤ端末)C1、C2およびC3と接続されている信号タグSignal Tag1、Signal Tag2およびSignal Tag3とラベルが付けられた(HARTデバイスからの)HART信号を有している。デバイス記述などのこれらのデバイスまたは信号に関するその他の情報も表示されてよい。

【0052】同様に、図2のProfibusマスタI/Oカード55に対応するProfibusカードは、ポートP01を介してそこに接続されているデバイスを有する。特に、Profibusデバイス(例えば、図1のProfibusデバイス50に対応するPBDE

V1だけが図6に示されている)は、Profibusカードを介して制御装置12に接続されている。各デバイスの下には、そのデバイス用のデバイス全体のパラメータが表示されてよく、デバイスに関連するスロットが表示されてよい。本発明に従って、各スロットは、デバイスに関連するモジュール用の位置設定記号であり、スロットは、モジュラーデバイス内のモジュールが静止スロットの間で動き回ってよいため、Profibusデバイスのために使用される。このようにして、モジュラーProfibusデバイスのケースでは、モジュールはスロットの間で移動されてよいが、スロット自体は固定されている。固定デバイスのケースでは、スロットはつねに同じモジュールを含むだろう。Profibusデバイスのモジュールが接続される各スロットの下で、Profibusまたはそのスロット内のモジュールに関連するスロットパラメータは、スロットないのモジュールと関連する信号とともに示されている。各信号は、信号名を含み、各信号の下で、その信号のDSTが示されている。DSTは、典型的には、構成システム70を介してユーザによって割り当てられ(るが、デバイス定義がsignal is createdを持つときに自動的に割り当てることができる)、制御装置12またはその他のデバイスによって、その信号がProfibusマスタI/Oデバイス55内のメモリロケーションから検索される、またはその中に配置されるときに信号を識別するために使用される。言うまでもなく、さらに多くの信号、スロット、パラメータ、デバイス、カード等が制御装置に取り付けられ、図6の階層に示されることが理解されるだろう。特に、図1のデバイス51と52の異なるProfibusデバイスエントリは、図5の階層内のポートP01に接続されているとして示されるだろう。同様に、デバイス51のデバイスエントリは、それぞれがそれに関連するパラメータ、信号、およびDSTを有する2つのスロットを含むだろうが、デバイス52のデバイスエントリは、それぞれが関連するパラメータ、信号およびDSTを有する3つのスロットを持つだろう。言うまでもなく、この情報のすべては、ユーザがシステムにデバイスを追加するとき、システムをセットアップするとき、あるいはそれ以外の場合システムを構成するときに図2のユーザ入力ルーチン74を介してユーザによって提供されてよく、構成データベース72に格納されている。

【0053】同様に、図1の制御装置12に接続されているASインタフェースデバイスネットワーク36は、例えば制御装置12に接続されているASインタフェースマスタI/Oカード60を含む。図1に図示されているように、このカードは、そこに接続されている4つのASインタフェースフィールドデバイスを有し、各デバイスはそこに関連しているASI 離散I/O入力および/または出力を有してよい(4つまで)。この情報は、

(ASDEV1と名付けられている) デバイス62-64の1つがASインタフェースカードのポートP01に接続されているとして、およびDSTがそれに関連付けられているInputD1とInputD2と呼ばれている2つのASI離散I/O信号を有するとして示されている図5Bの階層に示されている。言うまでもなく、図5と図6の文書階層を作成するために必要とされている情報は、構成データベース72に格納され、システム10の構成の前に入力ルーチン74を介するユーザ入力を通じて入手される。

【0054】図5と図6の階層に示されている構成情報は、ユーザ入力ルーチン74によって所望の方法で入手されてよいが、1つの実施態様においては、ユーザ入力ルーチン74は、ユーザに、ProfibusデバイスネットワークとASインタフェースデバイスネットワークのそれぞれの中で異なるデバイスに関する情報を入力するようにプロンプトを出すために図7から図26に示されている画面を使用してよい。所望される場合、デバイスによって作成され、構成データベース72内に格納されているデバイス定義内のその他の情報のどれかなどの図5と図6の階層の中のデバイスのそれぞれについてのその他の情報は、例えばユーザがデバイスを選択するとデバイスに関して表示されてよい。

【0055】それ以外の画面だけではなく図7から図26の画面も、それ以外のフォーマットも使用できるだろうが、標準ウィンドウズ型コマンドとともにウィンドウズ型フォーマットを使用して作成、修正されてよい。デバイス記述ファイルなどのGSDまたはその他の製造メーカファイルによって指定されるか、あるいは回数(times)とユーザに関係する情報などの情報のいくつかは、ユーザが変更可能ではなく、この情報はユーザ変更不可、つまり編集フィールドの外側として図7から図26の画面表示に示されている。画面表示、またはこれらのまたは類似した画面を作成するために必要とされる情報は、図2のテンプレート80と82に格納され、デバイスネットワーク34と36内のデバイスに関するデータを取得するためにユーザ入力ルーチン74によって使用されてよいことが理解されるだろう。以下に提供されている表は、さらに明確に、エントリのそれぞれにその情報がどのように入手されるのかだけではなく、各デバイスがデバイスネットワークを構成するため、およびそのネットワークを文書化するために取得されてよい情報の1つの考えられるフォーマットも指定する。しかしながら、この同じ情報またはその他の情報は、そのように所望される場合、それ以外のソースからだけではなく、それ以外のフォーマット、それ以外のデータ型等でも構成ルーチン70によって入手されてよいことが理解されるだろう。また、この情報のどれかまたはすべては、適切なデバイスのデバイス定義として格納されてよい。

【0056】図7から図16の画面表示は、Profibusネットワークまたはデバイス内の情報のエントリに関するが、図17から図26の表示は、ASインタフェースネットワークまたはデバイスの情報のエントリに関する。一般的には、ユーザ入力ルーチン74は、図5と図6の構成階層のような構成階層を表示し、その階層内の構成要素を選択してから、新しい要素を作成したり、ユーザ入力ルーチン74を使用してその階層内の既存の要素を編集できるように、文書化ルーチン78とともに使用されてよい。ユーザによって入力される、あるいはそれ以外の場合、例えば製造メーカファイルから入手される情報は、ProfibusデバイスまたはASインタフェースデバイスのためのデバイス定義を作成または更新するために使用されてよい。このようにして、例えば、ユーザは、Profibusデバイスの元の図5の階層内でファミリーを選択し、それに関連する新しい製造メーカ、モデル、改訂、モジュール等を入力し、一般的にはこの新しい要素についての情報を提供してよい。この時間中、ユーザ入力セクション74は、これらの要素に関する情報のすべてを入手するために、これらの要素のそれぞれに関連する画面のすべてをシーケンスで提供してよい。ユーザおよびGSDまたはその他のファイルによって提供されるような新しいファミリー、デバイス等は、それから、構成データベース72のライブラリ部分に格納される。同様に、ユーザは、階層内で指定されているポート、制御装置にシステム内で接続されている実際のデバイス、スロット、モジュール、信号等を指定または定義するために、図6のシステム構成セクションの下でProfibusカード、デバイス、スロット、モジュール、信号等を選択してよい。このようにして、システム構成セクションはシステムの実際の物理的な構成を文書化するが、ライブラリは一般デバイスに関するが、プロセス制御システム10内のこれらのデバイスの実際のまたは特定のインスタンスではない情報を格納する。図5および図6の階層は、ソフトウェア構成要素または制御装置等などのデバイス内に配置されてよい制御構成要素などの実際の構成要素を定義するシステム構成要素も有してよい。

【0057】例えば、ユーザが図5の階層内でProfibusデバイスのデバイス改訂を選択または作成すると、ユーザ入力セクション74は、自動的に、デバイス改訂オブジェクトダイアログの第1の一般ページを示す図7の画面を提供してよい。ここでは、ユーザは、デバイス改訂の記述を提供することができるが、ユーザ入力ルーチン74は、オブジェクトタイプ、識別番号、およびデバイス改訂のハードウェアリソースとソフトウェアリリースを入手するためにGSDファイルにアクセスする。図7および表1に示されているように、記述は、(画面の編集ボックスを介して)図2のユーザ入力セクション74を介してユーザによって提供されるが、情報

の残りは編集不可で、ユーザと変更が行われている時点に基づき、GSDファイルからまたは構成システム70から提供されてよい。以下の表1は、図7の画面表示内

の情報をさらに特定して定義する。

【0058】

【表1】

表1—デバイス改訂プロパティダイアログの一般ページ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	静的	該当なし	該当なし	Profibus DP デバイス改訂	構成要素タイプ
修正	静的	該当なし	該当なし	今日の日付	最後に修正された日付
修正者	静的	該当なし	該当なし	カレントユーザ	このオブジェクトを修正したユーザ
記述	編集	該当なし	該当なし	なし	構成要素の記述
インデント番号	静的	該当なし	該当なし	なし	GSDファイルからのデバイスインデント番号
ハードウェアリリース	静的	該当なし	該当なし	なし	GSDファイルからのハードウェアリリース文字列
ソフトウェアリリース	静的	該当なし	該当なし	なし	GSDファイルからのソフトウェアリリース文字列

【0059】次に、ユーザは、その場合特定のProfibusデバイス内でモジュールを定義するために使用されてよい、図8に示されているようなデバイス改訂プロパティダイアログを選択してよい。表2は、再び、ユーザ入力セクション74によって取得される情報をさら

に特定して定義し、その情報のすべてがデバイスのためのGSDファイルから入手できることが注記されるだろう。

【0060】

【表2】

表2—デバイス改訂プロパティダイアログのモジュールページ

名称	型	最小	最大	デフォルト	内容
型	静的	該当なし	該当なし	なし	デバイス型（コンパクトまたはモジュラー）
最大データ長	静的	該当なし	該当なし	0	1度のメッセージ交換でデバイスから読み取ることのできる、またはデバイスに書き込むことができるデータの最大量
最大入力長	静的	該当なし	該当なし	0	1つのメッセージ内でデバイスから読み取ることのできるデータの最大量
スロットの最大数	静的	該当なし	該当なし	0	デバイス内のスロットの最大数
最大出力長	静的	該当なし	該当なし	0	1つのメッセージでデバイスに書き込むことのできるデータの最大量
モジュール開始番号	静的	該当なし	該当なし	0	デバイス内の最初のモジュールがこの番号で始まる。

【0061】同様に、図9-15のウィンドウ表示は、ユーザ入力ルーチン74によって、それが、ユーザが新しいデバイス、または（プロセス制御システム10内でデバイスに関係する）階層内のデバイスの要素を指定しているのを認識するときにユーザに提供されてよい。言うまでもなく、信号名またはDSTなどのこの情報のいくらかは、デバイスまたは他の要素が図6の階層のシステム構成セクションに実際に配置されるとき、つまりシステム内で接続されている実際のデバイスに関する文書化が必要とされるときに、ユーザによって提供されてよい。

【0062】図9は、そのすべてが特定のデバイスに関してGSDファイルからユーザ入力ルーチン74によって入手できるボーレート、フェイルセーフ等に関する情報を含むデバイス改訂プロパティダイアログの上級ページを示す。表3は、このデータに関するさらに多くの情報を提供する。ユーザは、適切な画面を選択し、その編集フィールドを使用し、例えばGSDファイルからそれ以外の場合ユーザ入力ルーチン74が使用できないデータを入力することができる。このデータは、入力されると、デバイスのための構成データベース72に格納されてよい。

【0063】

【表3】

表3-デバイス改訂プロパティダイアログの上級ページ

名称	型	最小	最大	デフォルト	内容
サポートされているポーレート	静的	該当なし	該当なし	なし	デバイスによってサポートされているポーレートのリスト
サポートされているフェイルセーフ	静的	該当なし	該当なし	なし	デバイスによってサポートされているフェイルセーフである
サポートされている自動ポー検出	静的	該当なし	該当なし	0	デバイスによってサポートされている自動ポー検出である
最小スレープ間隔	静的	該当なし	該当なし	0	数百マイクロ秒単位の最小スレープ間隔

【0064】図10は、特定のデバイス改訂のためのデバイス改訂パラメータを定義または記述するために使用されるデバイス改訂パラメータダイアログボックスのページである。言うまでもなく、類似したボックスは、Profibus デバイスのモジュールパラメータまたはスロットパラメータを定義するために使用できるだろう。パラメータの実際の型は、例えば、整数、自然数、実数、列挙値、16進データ、任意のサイズまたは次元の配列等を含む所望される型であってよい。以下の表

4は、整数の定義、列挙パラメータおよび16進パラメータに関する情報を提供するが、図11は、値フィールドがマイクロソフト (Microsoft) によって提供されているMSグリッド制御などのグリッド制御を使用するPARAM3と名前が指定される16進データパラメータを示す。

【0065】

【表4】

表4-パラメータプロパティ

名称	型	最小	最大	デフォルト	内容
パラメータ名	編集	該当なし	該当なし	なし	パラメータの名前
パラメータ	結合	該当なし	該当なし	なし	パラメータの型
値	編集 MS グリッド 制御	該当なし	該当なし	なし	パラメータの値 (整数、実数、列挙、16進等)

【0066】図12は、Profibus DPモジュールを作成または編集するために使用される画面を示すが、表5はこのようなモジュールに定義されるプロパティのリスティングを提供する。再び、静的型の設定されるフィールドのそれぞれは、デバイスのためのGSDフ

ァイルから入手されるか、あるいはシステムの現在の稼働条件 (時間、ユーザ等) に基づいてよい。

【0067】

【表5】

表 5 - D P モジュールプロパティダイアログ

名称	型	最小	最大	デフォルト	内容
オブジェクト型	静的	該当なし	該当なし	Profibus DP モジュール	構成要素型
修正	静的	該当なし	該当なし	今日の日付	最後に修正された日付
修正者	静的	該当なし	該当なし	カレント ユーザ	このオブジェクトを修正 したユーザ
記述	編集	該当なし	該当なし	なし	構成部品の記述
入力データ長	静的	該当なし	該当なし	なし	入力メッセージ内のバイ ト数
出力データ長	静的	該当なし	該当なし	なし	入力メッセージ内のバイ ト数
パラメータデー タ長	静的	該当なし	該当なし	なし	パラメータデータ内のバ イト数
識別子	静的	該当なし	該当なし	なし	サイズが 1 から 17 バイ トである識別子のリスト

【0068】同様に、図13は、ユーザがモジュールまたはスロット内、あるいはモジュールまたはスロット内のProfibus信号を作成または編集できるようにする画面を示すが、表6はこのような信号のプロパティに関する情報を提供する。各信号は、専用のDSTを有するだろう。一般的には、次に示すデータを、Profibusデバイス信号ごとに指定することができる。

1) 信号方向、つまりプロセス入力または出力。双方向信号は、通常、別個のDSTとして構成できる。2) サポートされているデータ型のセットを含む信号値のデータ型、および適切な場合、信号に関してサポートされているビッグエンディアンフォーマットおよびリトルエンディアンフォーマット（つまり、それが複数バイト信号であるときに信号のバイト順）。3) モジュールに／モジュールから提供されるデータ文字列内の信号値のロケーション。典型的には、この情報は、バイトオフセット、場合によってはバイト内の追加ビット数のフォーマットを取ることもできる。信号を含むモジュールがそれと関連する複数のProfibus-DP識別子を有す

る場合、バイトオフセットは、1つの特定の識別子ではなく、モジュール全体に関してよい。4) 信号の名称。これは（図6の階層などの）階層に表示される名称であり、信号を用いて制御を実行するときに、例えば制御装置12によって使用される特定の信号のタグであるDSTとは別個である。該名前は、可変速度駆動装置などの複雑なデバイスからの信号に特に有効である。5) Profibus-DPチャンネル番号。この情報は、診断および信号ステータス生成に有効であるが、通常、デバイス診断サポートを持たない可能性のある信号があるため、オプションである。信号を含むモジュールがそれに関連する複数の識別子を持つ場合、このモジュール内の識別子の位置（第1、第2、第3等）も、Profibus-DPプロトコルによって指定される診断メッセージが識別子／チャンネルに基づいた診断情報を供給するため、指定されなければならない。

【0069】

【表6】

表 6 - 信号プロパティダイアログ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	静的	該当なし	該当なし	Profibus信号	構成要素タイプ
修正	静的	該当なし	該当なし	今日の日付	最後に修正された日付
修正者	静的	該当なし	該当なし	カレントユーザ	このオブジェクトを修正したユーザ
記述	編集	該当なし	該当なし	なし	構成要素の記述
データフォーマット	結合	該当なし	該当なし	ビッグエンディアン	ビッグエンディアンフォーマットまたはリトルエンディアンフォーマットのどちらか
信号方向	結合	該当なし	該当なし	入力	入力または出力のどちらか
データ型	結合	該当なし	該当なし	整数 8-ビット (Int8)	Int8, Int16, Int32, UInt8, UInt16, UInt32, ビットフィールド
バイトオフセット	スピン 数値	0	255	0	メッセージの始まりからの信号のバイトオフセット
開始ビット	スピン 数値	0	16	0	データ型がビットフィールドである場合には、この編集が可能にされ、それはフィールドが開始するメッセージの開始からのビット内のオフセットを表す
ビットの数	スピン 数値	0	16	1	ビット内のビットフィールドのサイズ
診断チャンネル	結合	0	255	なし	診断チャンネル番号
識別子インデックス	スピン 数値	0	255	0	診断チャンネルが属する識別子のインデックス

【0070】言うまでもなく、同じモジュールの複数のインスタンスのために信号記述を再利用することが望ましい。その結果、ユーザが（図5のライブラリなどの）デバイス型ライブラリ内の特定のモジュールのためのデフォルトとして信号のリストを構築してから、モジュールインスタンスが階層のシステム構成部分で構成されるときに信号記述を修正する（あるいは信号を追加または削除する）ことができるようにすることが望ましい場合がある。

【0071】ユーザは、デバイスプロパティを入力することによってデバイスのプロパティを作成または編集してもよい。ユーザは、ポートの下でデバイスを選択するか、あるいはポートに接続されているデバイスをインストールしてから、前記に示されたダイアログなどのプロパティデバイスダイアログを入力してよい。デバイスプロパティは、アドレスフィールド、ウォッチドッグタイマイネーブル、および使用可能にされている場合、ウォッチドッグタイマの値が追加されたデバイス改訂プロパティと同じである。ここでは、Profibusカードは、構成データベース内でインストールされ、作成された（または、自動検出された）と仮定されている。そうである場合、ユーザは、カードの下のポートを選択して

から、そのポートに接続されている新規デバイスのデータを入力することができる。構成システム70は、デバイス改訂、製造メーカ等を要求してよく、デバイスのGSDファイルからなどのデバイスに関するデータが入手できる場合、新規デバイスが画面に表示され、ユーザは新規デバイスに使用できないデータを入力するようにプロンプトを出される。デバイスがポート設定値と互換性がない場合、デバイスはインストール可能ではない可能性があるため、ユーザは通知されてよい。デバイス改訂がコンパクトデバイスである場合には、すべての必要なスロットおよびパラメータは、GSDファイル情報に基づき、即座にデバイスの下で作成される。デバイスアドレスは、所望される場合に次に使用可能なアドレスに設定されてよい。言うまでもなく、ユーザは所望されるようにデバイスのプロパティを編集してよい。

【0072】デバイスがモジュラーデバイスである場合、ユーザは、デバイスと関連する1つまたは複数のスロットを作成してよい。構成システム70は、ユーザに（テンプレート80または製造メーカファイル88によって提供されるような）スロット内で使用するためのモジュール名のリストを提供してよい。ユーザは、スロット数の限度までの任意の数のスロット、総入力長、総出

力長、総長および総パラメータ長を作成する。複数の識別子を含むスロットのサイズは、識別子の合計および（すべてのモジュールに適用する）構成サイズに対する制限によって決定される。図14および図15は、モジュラーデバイス用のモジュールおよびスロットの作成または編集と関連する画面を示すが、以下の表7と8はモジュールまたはスロットの異なるプロパティに関する情報を提供する。Profibusコンフィギュレータ

表7-DPモジュールブラウザダイアログ

名称	型	最小	最大	デフォルト	内容
モジュールリスト	リスト 制御	該当 なし	該当 なし	なし	各モジュールの入力長と出力長だけではなくデバイス型定義からのモジュールのリスト
バイトが使用されている表	12ビット	該当 なし	該当 なし	なし	どのくらいの数のバイトが使用されているのか、およびスロットだけではなくどのくらいの数が残っているのかを示す表

【0074】

【表8】

表8-スロットプロパティダイアログ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	静的	該当 なし	該当 なし	Profibus スロット	構成要素タイプ
修正	静的	該当 なし	該当 なし	今日の日付	最後に修正された日付
修正者	静的	該当 なし	該当 なし	カレント ユーザ	このオブジェクトを修正したユーザ
記述	編集	該当 なし	該当 なし	なし	構成要素の記述
使用可能	チェック ボックス	該当 なし	該当 なし	真	スロットは使用可能にされるか（それはダウンロードされるだろうか）
入力データ長	静的	該当 なし	該当 なし	なし	入力メッセージ内のバイト数
出力データ長	静的	該当 なし	該当 なし	なし	出力メッセージ内のバイト数
パラメータデータ長	静的	該当 なし	該当 なし	なし	パラメータデータ内のバイト数
識別子	静的	該当 なし	該当 なし	なし	識別子のリスト
モジュール番号	編集	モジュール 番号1	モジュール 番号255	モジュール開 始番号	ダウンロード時にこのスロットのために使用するモジュール番号

【0075】言うまでもなく、ユーザは、所望されるようにスロットを作成、使用可能、使用禁止または削除してよい。ユーザがスロットを使用禁止にすると、モジュール番号は空白に設定され、モジュール名は保持され、モジュール番号シーケンスにギャップが生じるため、ユーザは、ギャップがないように、別のスロットにモジュール番号を割り当てるように自動的に要求される。それ以外の場合、割り当てられたスロットのためにモジュールを使用するダウンロードは成功しない可能性がある。スロットを使用可能にするとき、モジュール番号は次に

76は、Profibus-DPデバイス内のモジュールの順序が、カレントProfibus-DP構成ツールでのケースである、デバイス全体を構成し直さなくても配列し直すことができるように特定のDPモジュールインデックスを指定するだろう。

【0073】

【表7】

使用可能な未使用モジュール番号に設定されてよい。スロットを削除すると、モジュールは別のスロットで使用するために利用できるようになる。

【0076】ユーザは、スロットの下で1つまたは複数の信号を作成してもよい。このケースでは、信号は、スロットが存在するデバイスのために作成され、ユーザが信号に名前を指定すると、その信号にDSTが作成される。このDSTは、リモートI/Oネットワーク内のデバイスから生じる信号を識別するために、制御装置12によって使用されてよい。

【0077】言うまでもなく、ユーザは、デバイス、信号、スロット等をコピーし、これらのコピーされたデバイスを新規デバイス、信号、スロット等の作成のために使用してよい。ユーザは、ポート定義も作成または編集してよい。図16は、(P01と名前が付けられてい

る)ポートを定義するために使用される画面を示すが、表9はポートのプロパティを定義する。

【0078】

【表9】

表9-ポートプロパティダイアログ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	静的	該当なし	該当なし	Profibus DP スロット	構成要素タイプ
修正	静的	該当なし	該当なし	今日の日付	最後に修正された日付
記述	編集	該当なし	該当なし	カレント ユーザ	このオブジェクトを修正 したユーザ
ポーレート	結合	該当なし	該当なし	なし	構成要素の記述
使用可能	チェック ボックス	該当なし	該当なし	未チェック	使用可能にされたポート
アドレス	結合	該当なし	該当なし	0	ポートアドレス0-256

【0079】また、構成システム70は、ユーザの動作に基づき、信号、デバイス、モジュール情報等の情報がいつ必要とされるのかを認識し、ユーザに、たとえばここに示されているダイアログ画面を使用して、この情報を入力するように自動的にプロンプトを出してよい。

【0080】存在するProfibus-DPデバイスの数およびプロトコルとインプリメンテーションの相対的な成熟度を考慮すると、ユーザが準拠するデバイスをネットワークに接続し、該ネットワークをデバイスのデータ交換するように構成できることが望ましい。これらの動作を達成するために、以下のステップが、図2の構成システム70によって実行できる。第1に、ユーザはGSDファイルをインポートする必要がある、ユーザ入力セクション74はこのインポートフォーマットをインポートサブメニューオプションとして使用できるようにする。GSDファイルのインポート(Importation)は、スレーブファミリー(本来、GSD仕様の中に一覧表示されている標準化されているデバイスタイプ)、ベンダ、モデルおよび改訂の階層の下を図5のライブラリ内でデバイス定義を自動的に作成することができる。

【0081】第2に、ユーザは、ライブラリ内の任意のデバイスのために、リストモジュール、および任意のモジュールのための信号を構築し、デバイスまたはモジュールパラメータのデフォルト値を指定することもできる。ユーザ入力ルーチン74は、ユーザが、これを自動的にできるようにしてよい。これらのパラメータ値は、GSDファイルに供給されているデフォルト値を無効にする。第3に、ユーザは、つまりProfibusデバイスの制御システム10内のProfibusマスタI/Oカードへの実際の接続を反映するために、例えば、図6の階層内の適切なProfibus I/Oカード

ポートの上でデバイスを作成(またはデバイスをそれに接続)する。これは、ユーザ入力ルーチン74とともに文書化ルーチン78を使用して達成されてよい。ユーザは、デバイスのアドレスを指定することができ、それがモジュラーデバイスである場合には、デバイスが含むモジュールの正確な順序で指定することができる。各モジュールは、信号とパラメータ値がデバイスライブラリ内のデバイス型に指定された状態で作成されるが、ユーザはこれらの値を無効にし、信号を追加または削除することができる。ただし、ユーザは、信号にすべてのDSTを割り当てる必要があり、これらのDSTはプロセス制御ルーチンを実行するために制御装置12によって使用されている。

【0082】第4に、ユーザは、Profibus I/Oカード用コンフィギュレータ76を使用して、I/Oカード上のポートをProfibus I/Oカードにダウンロードする。構成データが正しく、構成が一致する場合は、デバイスおよびI/Oカード55はデータ交換を開始する。第5に、ユーザは、制御アプリケーション内の信号ごとにDSTを指定することによって信号を使用するために制御アプリケーションを構成する。DST用の実際の経路は、デバイス、スロット、および信号の順であってよく、制御装置モジュール(つまり、Profibusカードからの信号を使用する制御装置12内のソフトウェアモジュール)は、新規ProfibusデバイスまたはProfibusモジュールが追加される場合にはダウンロードし直される必要がないことを意味する。Profibusカードの再構成は、ユーザがスロット内で信号を配列し直したり、デバイスのアドレスを変更する場合に実行されさえすればよい。言うまでもなく、ユーザは、ASインタフェースデバイスネットワーク、あるいはFieldbusデバイスネットワ

ークおよびHARTデバイスネットワークを含む任意のそれ以外のデバイスネットワークに類似したステップを実行することができる。

【0083】ユーザがデバイス型ライブラリ内のデバイスのデフォルトパラメータ値を変更する場合、すでに作成されているデバイスのパラメータは、好ましくは影響を受けないだろう。デバイス型ライブラリ内のデバイスのモジュール用のパラメータまたは信号が変更される場合、そのモジュールのカレントインスタンスは影響を受けないが、既存のデバイスインスタンスの場合であっても作成される将来のモジュールインスタンスは、新規パラメータ値および信号を継承するだろう。モジュール信号およびパラメータのライブラリからデバイスインスタンスへの継承を可能とするために、ライブラリ定義がすでに作成されているデバイスと一貫したままであることを保証することが必要である。したがって、デバイスインスタンスがライブラリ内の特定の改訂と関連付けられている場合、その改訂はインポートされ直したり、削除される必要はない。

【0084】ProfibusデバイスのGSDファイルは、デバイスから読み取られる診断メッセージ内に提供されている特定のビットおよびエラーコードのテキスト記述を含む。したがって、構成アプリケーションは、この情報をユーザに提示するためにデバイス定義へのアクセスを持たなければならない。GSDファイルからのすべての情報は構成データベース72内のデバイス用オブジェクトに格納できるため、このアクセスは構成データベース72を使用して達成できる。

【0085】Profibusリンク53上でのProfibusスレーブデバイスアドレスの割当ては、一度に1つのデバイスに制限され、この割当てはすべてのスレーブデバイスによってサポートされていない。それでもなお、アドレス割当ては、所望される場合、このカードが構成されるときにProfibus I/Oマスタカード55に適切なソフトウェアを提供することによって、サポートすることができる。また、所望される場合、自動検出ソフトウェアは、デバイスの構成の部分としてProfibusマスタI/Oデバイス55に配置

されてよく、このソフトウェアは、多くのケースでは特定のモジュールを決定することはできないが、Profibusスレーブデバイスを自動検出するために使用されてよい。所望される場合、自動検出ソフトウェアは、Profibusバス53上でアドレスをポーリングし、それらのアドレスにあるデバイスを検出するために動作できる。デバイスが検出されると、デバイスの存在は、構成データベース72内のエントリとして配置される構成ルーチン70に送信され、文書化ルーチン78によって作成される階層を介してユーザに表示される。ユーザが検出されたデバイスを表示すると、ユーザは、例えば、Profibusテンプレートデータベース80内の適切なテンプレートを使用してそのデバイスに関する情報を提供するように、ユーザ入力ルーチン74によってプロンプトを出されてよい。Profibusプロトコルでは、パラメータ値をアップロードすることはできず、信号構成を決定することはできない。しかしながら、所望される場合、ProfibusマスタI/Oデバイス55は、デバイスの構成をアップロードし、そのデバイスが、矛盾点を検出し、これらの矛盾点を、ワークステーション14を介してユーザに表示するために構成データベース72内でどのように構成されるのかとの比較を提供するようにプログラミングされてよい。

【0086】類似する方法で、ユーザは、そのネットワーク内のデバイスからの信号を使用する、およびそのネットワーク内のデバイスに信号を送信するためだけではなく、ASインタフェースネットワーク36を構成するためにも、ASインタフェースデバイスのデバイス定義を作成するためにASインタフェースデバイスのそれぞれに関する情報を入力してよい。例えば、ユーザは、例えば図17のデバイスタイププロパティ画面（一般情報）、図18（プロファイルページ）、図19（入力ページ）、図20（出力ページ）、および図21（パラメータページ）を使用して、ASインタフェースデバイス型を定義してよい。表10から14は、これらのページ内のプロパティを定義する。

【0087】

【表10】

表10-ASインタフェースデバイス型ダイアログの一般プロパティページ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	編集	該当なし	該当なし	ASインタフェースデバイス	構成要素タイプ
修正	編集	該当なし	該当なし	今日の日付	最後に修正された日付
修正者	編集	該当なし	該当なし	カレントユーザ	このオブジェクトを修正したユーザ
記述	編集	該当なし	該当なし	なし	構成要素の記述

【0088】

【表11】

表 11-ASインタフェースデバイス型ダイアログのプロファイルプロパティページ

名称	型	最小	最大	デフォルト	内容
プロファイル	編集	該当なし	該当なし	S-0-0	S-[I/O構成]-[識別コード]の組み合わせ
I/O構成コード	結合ボックス	該当なし	該当なし	0x0 IN IN IN IN IN	対応する記述子のあるI/O構成タイプ0x0から0xF—以下の注記を参照のこと。
識別コード	結合ボックス	該当なし	該当なし	0x0	識別コード0x0から0xF

【0089】注記：I/O構成結合ボックスは、以下のリストを含んでよい。

【0090】“0x0 IN IN IN IN”,
 “0x1 IN IN IN OUT”,
 “0x2 IN IN IN I/O”,
 “0x3 IN IN OUT OUT”,
 “0x4 IN IN I/O I/O”,
 “0x5 IN OUT OUT OUT”,
 “0x6 IN I/O I/O I/O”,
 “0x7 I/O I/O I/O I/O”,
 “0x8 OUT OUT OUT OUT”,
 “0x9 OUT OUT OUT IN”,
 “0xA OUT OUT OUT I/O”,
 “0xB OUT OUT IN IN”,

“0xC OUT OUT I/O I/O”,

“0xD OUT IN IN IN”,

“0xE OUT I/O/ I/O I/O”,

“0xF TRI TRI TRI TRI”,

ASインタフェースプロトコル内で使用可能な設定値により提供され、それらに対応するこれらの設定値は、それに従ってI/O構成選択が行われる入力プロパティページと出力プロパティページ上でチェックボックスを使用可能/使用禁止にするために使用できる。例えば、0x0が選択されると、すべての出力チェックボックスは使用禁止されるだろう。

【0091】

【表12】

表 12-ASインタフェースデバイス型ダイアログの入力プロパティページ

名称	型	最小	最大	デフォルト	内容
I 1	チェックボックス	該当なし	該当なし	未チェック	D0入力使用可能
I 1	結合ボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの入力離散I/Oのラベルを入力できる。
I 2	チェックボックス	該当なし	該当なし	未チェック	D1入力使用可能
I 2	結合ボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの入力離散I/Oのラベルを入力できる。
I 3	チェックボックス	該当なし	該当なし	未チェック	D2入力使用可能
I 3	結合ボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの入力離散I/Oのラベルを入力できる。
I 4	チェックボックス	該当なし	該当なし	未チェック	D3入力使用可能
I 4	結合ボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの入力離散I/Oのラベルを入力できる。

【0092】

【表13】

表 1 3 - A S インタフェースデバイス型ダイアログの出力プロパティページ

名称	型	最小	最大	デフォルト	内容
O 1	チェックボックス	該当なし	該当なし	未チェック	D 0 出力使用可能
O 1	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの出力離散I/Oにラベルを入力できる。
O 2	チェックボックス	該当なし	該当なし	未チェック	D 1 出力使用可能
O 2	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの出力離散I/Oにラベルを入力できる。
O 3	チェックボックス	該当なし	該当なし	未チェック	D 2 出力使用可能
O 3	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの出力離散I/Oにラベルを入力できる。
O 4	チェックボックス	該当なし	該当なし	未チェック	D 3 出力使用可能
O 4	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこの出力離散I/Oにラベルを入力できる。

【 0 0 9 3 】

【 表 1 4 】

表 1 4 - A S インタフェースデバイス型ダイアログのパラメータプロパティページ

名称	型	最小	最大	デフォルト	内容
P 0	チェックボックス	該当なし	該当なし	未チェック	P 0 パラメータ使用可能
P 0	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこのパラメータのラベルを入力できる。
P 1	チェックボックス	該当なし	該当なし	未チェック	P 1 パラメータ使用可能
P 1	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこのパラメータのラベルを入力できる。
P 2	チェックボックス	該当なし	該当なし	未チェック	P 2 パラメータ使用可能
P 2	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこのパラメータのラベルを入力できる。
P 3	チェックボックス	該当なし	該当なし	未チェック	P 3 パラメータ使用可能
P 3	チェックボックス	該当なし	該当なし	使用禁止	チェックボックスがチェックされたら、使用可能にする。するとユーザはこのパラメータのラベルを入力できる。

【 0 0 9 4 】 ユーザは、新しい A S インタフェースカードを作成、編集することもできる。ユーザは、例えば図 6 の中に示されている文書化ルーチン 7 8 によって作成

されている階層内のコンテキストメニューから新規カードを選択し、システムに新しい A S インタフェースカードを作成させる A S インタフェースとしてカードタイプ

を選択してもよい。同様に、ユーザは、ASインタフェースデバイス、および関連する入出力とパラメータを追加することによって階層内でASインタフェースポートを作成、編集等してよい。このようなポートのプロパティ画面は図22に示され、プロパティは以下の表15に定義されている。

【0095】

【表15】

表15-ASインタフェースポートプロパティダイアログ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	編集	該当なし	該当なし	ASインタフェースポート	構成要素タイプ
修正	編集	該当なし	該当なし	今日の日付	最後に修正された日付
修正者	編集	該当なし	該当なし	カレントユーザ	このオブジェクトを修正したユーザ
記述	編集	該当なし	該当なし	なし	構成要素の記述
使用可能	チェックボックス	該当なし	該当なし	未チェック	ポート使用可能フラグ
リセットデバイス	チェックボックス	該当なし	該当なし	チェック済み	制御装置が故障したときデバイスをリセットする
ポーリング続行	チェックボックス	該当なし	該当なし	未チェック	制御装置が故障したときにポーリングを続行する
自動アドレスインデックス	チェックボックス	該当なし	該当なし	チェック済み	ポートはアドレスを新規デバイスに自動的に割り当てるだろう。

【0096】同様に、ユーザは、ASインタフェースデバイスを作成または編集してよい。ユーザは、ASインタフェースポートを選択してから、コンテキストメニューから新規デバイスを選択してよい。例えば、そのプロパティが以下の表16と17にさらに詳細に定義されている、図23と23に示されているものなどの一般ペー

ジとプロパティページを有するASインタフェースデバイスプロパティダイアログウィンドウが自動的に表示される（つまり、それはユーザ入力ルーチン74によって作成される）。

【0097】

【表16】

表16-A Sデバイスプロパティ一般ページ

名称	型	最小	最大	デフォルト	内容
オブジェクトタイプ	編集	該当なし	該当なし	ASインタフェースポート	構成要素タイプ
修正	編集	該当なし	該当なし	今日の日付	最後に修正された日付
修正者	編集	該当なし	該当なし	カレントユーザ	このオブジェクトを修正したユーザ
名前	編集	該当なし	該当なし	デフォルトデバイス名 ASDEV1	デバイス名—ユーザが新しい子を選択した場合、使用可能、ユーザが修正した場合、使用禁止
記述	編集	該当なし	該当なし	なし	構成要素の記述
使用可能	チェックボックス	該当なし	該当なし	未チェック	ポート使用可能フラグ
デバイス型	プルダウン	該当なし	該当なし	ドロップダウンリストの最初のアイテム	デバイス型が作成されている—ユーザが新しい子を選択した場合、使用可能、ユーザが修正する場合使用禁止
新規タイプ追加	チェックボックス	該当なし	該当なし	該当なし	新規デバイス型の追加を可能にする。
プロファイル	静的	該当なし	該当なし	空	デバイス型のプロファイル
アドレス	チェックボックス	1	31	1	デバイスのアドレス

【0098】

【表17】

表17-A Sデバイス上級プロパティページ

名称	型	最小	最大	デフォルト	内容
Param 1	チェックボックス	該当なし	該当なし	チェック済み	param 1のビット値。デバイス型から名前を取る。
Param 2	チェックボックス	該当なし	該当なし	チェック済み	param 2のビット値。デバイス型から名前を取る。
Param 3	チェックボックス	該当なし	該当なし	チェック済み	param 3のビット値。デバイス型から名前を取る。
Param 4	チェックボックス	該当なし	該当なし	チェック済み	param 4のビット値。デバイス型から名前を取る。

【0099】ユーザは、デバイスのタグまたは名前を入力し、デバイスが使用可能にされているかどうかを示し、デバイスのデバイス型およびデバイスのアドレスを示してよい。デバイス型に関連している離散デバイス1/O構成要素は、ASインタフェースデバイスに関するテンプレート情報、またはASインタフェースデバイス用製造ファイルに基づいてこのデバイスのために自動的に作成され、デバイスには（デバイスの作成時、または図6の階層内のポートへのデバイスのダウンロード時にこの情報を入力するように自動的にプロンプトが出される）ユーザによってDSTが与えられるだろう。DSTは、それから、デバイスおよびデバイスから信号を識別するために制御装置12によって使用できる。

【0100】ASインタフェースデバイスは、バス上で

アドレス指定をサポートする必要があるが、一般的にはこれは当てはまる。その場合、アドレスは制御装置12内の不揮発性メモリまたはASインタフェースマスタ1/Oカード60内に格納されなければならない。アドレスは少なくとも10回再割当てすることができるのが好ましい。ASインタフェースプロトコル内には1つのデフォルトアドレス（ゼロ）しかないため、ユーザは、デバイスのアドレスをオフラインで割り当ててから、デバイスをネットワーク36に接続したり、アドレスゼロのデバイスを割り当て、ASインタフェースプロトコルによって提供されているこのデバイスの自動アドレス割り当て機能を使用して、ASインタフェースマスタ1/Oデバイス60を通してアドレスを割り当ててを選択してよい。

【0101】ASインタフェースデバイスの自動検出は、ASインタフェースカードの自動検知に類似した方法で実行されてよい。特定のポートに対する図6の階層からのメニューオプション時に、アドレス、I/O構成、および識別コードを含む検出されたスレーブデバイスのリストが、ASインタフェースマスタI/Oデバイス60から読み取られる。それから、ユーザは、ASインタフェーステンプレートデータベース82に格納されている例えば図17から23のテンプレートを使用して、自動的に要求されるその型とパラメータビット値を指定することによって、システム構成にこれらのデバイスを含むことができる。それから、構成ルーチン70はポートをダウンロードし、そのポートに接続されている新しいデバイスを起動してよい。また、自動検出ダイアログは、ユーザが、検出されたスレーブのアドレスをクリアできるようにし、この機能を実行するために携帯端末を使用する必要性を削除する。

【0102】ユーザは、また、所望される場合、ASインタフェースカード上でデバイスを自動検出してもよい。これを行うためには、ユーザはコンテキストメニューを表示させる(P01などの)ASインタフェースポートを選択してよい。ユーザは、コンテキストメニュー内で自動検出機能を選択してよい。システム70は、それから、ASインタフェースマスタI/Oカード60の自動検出機能を実現することによってデバイスを自動検出し、ユーザに、検出され、アドレスごとにデータベース内でデバイスに相互参照されたデバイスのリストを提供するだろう。このような画面表示は図25に示されている。データベース内にないデバイスは、図25のアドレス番号1、4および5にあるデバイスなどの、構成された欄の中に空白のエントリを有するだろう。ユーザは、未構成デバイスを選択し、構成ボタンをクリックし、そのデバイスを構成してよい。ユーザ入力ルーチン74は、それからユーザに、ユーザがデバイスを構成するために使用できるASインタフェースデバイスプロパティダイアログを提供する。ASデバイスプロパティダイアログボックスは、上級ページに入力されたプロファイルとアドレスとともに、名前フィールドにデフォルト名を含むだろう。プロファイルと互換性のあるデバイス型だけが、デバイス型結合ボックスの中に取り込まれるだろう。自動検出されたデバイスがアドレス0にある場合には、そのアドレスフィールドは選択可能となるだろう。それ以外の場合、アドレスフィールドは、好ましくは変更可能ではない。上級ページ(図示されていない)においては、構成システム70が、そのデバイス型に定義されているパラメータに関連するパラメータを取り込むだろう。これらのパラメータのいくつかは、すべてのパラメータが有効でない場合にもディスプレイされてよい。所望の変更が加えられた後に、ユーザは、デバイスを作成させ、構成データベース72内で構成させるO

Kボタンを選択する。依然として、コンフィギュレータ76は、ASインタフェースネットワーク36内のさまざまなデバイスによって使用されている複数のアドレスまたは同じアドレスを検出するなどのアドレス割当ておよびクリーニングを実行することができ、ユーザに検出された冗長アドレスを通知してよい。

【0103】ユーザは、ASインタフェースデバイスの離散I/Oプロパティを作成または指定してもよい。ユーザは、図5および図6の概略図などの文書概略図上でデバイスの内容区画内の離散I/O構成要素を選択してよい。コンテキストメニューが表示され、ユーザはプロパティ選択肢を選択してよい。図26のダイアログなどの離散I/O構成要素プロパティダイアログが表示され、ユーザは離散I/O構成要素の記述を提供してよい。

【0104】言うまでもなく、所望される場合、これらの種類またはその他の種類の遠隔I/Oネットワークに関するその他の情報も、デバイスネットワークの機能および設計に応じて提供できるだろう。さらに依然として、ユーザは、各デバイス内の各デバイス、モジュール、スロット、プロパティ、パラメータ等に関連する情報を所望の方法で入力、編集することができる。ただし、好ましくは、ユーザはシステムの使いやすさを可能にするために必要とされる情報を入力するようにプロンプトを出される。言うまでもなく、図2のテンプレートが画面表示、または画面表示を作り、デバイスネットワークのデバイス、デバイス型、信号、モジュール、パラメータ、スロット等のそれぞれのためにそこにデータに記入するために必要とされる情報を含むだろう。

【0105】構成ルーチン70は、単一データベース72内のシステム10に関連するさまざまなデバイスネットワークのそれぞれの中でデバイスのそれぞれを構成するために必要な情報のすべてを収集して格納し、遠隔I/Oデバイスネットワークを構成し、この構成を文書化するためにこの同じデータベースを使用するため、ユーザは、リモートI/Oネットワークに関するデータを1度入力するだけでよく、それは、ユーザが他の従来のデバイスI/OネットワークまたはローカルデバイスI/Oネットワークだけではなく他のリモートI/Oデバイスネットワークなどのそれ以外のデバイスネットワークに関する情報を入力すると同時に実行できる。このデバイス情報のすべては、さまざまなデバイスが、ローカルデバイスネットワーク、専用デバイスネットワーク、および遠隔I/Oデバイスネットワークと関連するマスタデバイスを構成するためだけではなく、プロセス制御システム10内の制御装置を通してどのように接続されるのかを文書化するためにも使用できる1つの共通した統合構成データベース72の中に格納される。

【0106】コンフィギュレータ76をホストワークステーション12(もしくは制御装置12)に設置し、構

成データベース72に格納されているデータを使わせることでユーザは各々の遠隔I/Oネットワーク(ネットワーク34や36のような)の各デバイスに関する情報を1度だけ入力することができるが、この情報はFieldbusやHARTデバイスなどのシステム10内の他のデバイスに関する構成情報と統合され、システム10内の他のデバイスに関する情報が図5および図6のようなエクスプローラ型ツリーの階層に文書化されるのと同様に自動文書化することができ、遠隔I/Oネットワークデバイスが自動的に構成される。

【0107】構成ツール70は、FieldbusデバイスおよびHARTデバイスとともに使用されていると記述されてきたが、それは任意のほかの外部プロセス制御デバイス通信プロトコルも構成、文書化するために実現することができる。ここに説明されている構成ツール70は、好ましくはソフトウェア内で実現されるが、それはハードウェア、ファームウェア等で実現されてよく、プロセス制御システム10と関連するその他のプロセッサによって実現されてよい。したがって、ここに説明されているルーチン70は、標準多目的CPU内で、または所望されるように、特に設計されたハードウェアまたはファームウェア上で実現されてよい。ソフトウェア内で実現されると、ソフトウェアルーチンは、磁気ディスク、レーザディスクまたはその他の記憶媒体上、コンピュータまたはプロセッサのRAMまたはROM内などの任意のコンピュータ読取り可能メモリで格納されてよい。同様に、このソフトウェアは、例えばコンピュータ読取り可能ディスクまたはその他のトランスポート可能なコンピュータ記憶装置機構上、あるいは(トランスポート可能な記憶媒体を介してこのようなソフトウェアを提供することと同じまたは交換可能であるとして表示されている)電話回線、インターネット等の通信路上でを含む既知のまたは所望される送達方法を介してユーザまたはプロセス制御システムに送達されてよい。

【0108】したがって、本発明は、例示的であることだけが意図され、発明の制限的でないことが意図される特定の例に関して説明されてきたが、変更、追加または削除が、本発明の精神または範囲から逸脱することなく開示されている実施態様に対して行われてよいことは、普通の技術の熟練者にとって明らかだろう。

【図面の簡単な説明】

【図1】ローカルI/O、専用I/O、および遠隔I/Oデバイスネットワークに接続されている制御装置を有するプロセス制御システムのブロック図である。

【図2】プロセス制御システム内でローカルI/Oデバイスネットワーク、専用I/Oデバイスネットワークおよび遠隔I/Oデバイスネットワークを構成するために、ローカルI/Oデバイスネットワーク、専用I/Oデバイスネットワーク、および遠隔I/Oデバイスネットワークに関して、ユーザから情報を受け入れる構成シ

ステムのブロック図である。

【図3】図1のプロセス制御システムのプロセス制御構成システムで使用される信号オブジェクトデータベースの一部のブロック図である。

【図4】図1のマスタI/Oデバイスで使用される共用メモリのブロック図である。

【図5】Profibus I/O通信プロトコルおよびASインタフェースI/O通信プロトコルを介して制御システム内で接続されているデバイスの文書化および構成を、FieldbusおよびHART通信プロトコルを介して制御システム内で接続されているデバイスと統合する図2の構成システムと関連する構成文書化階層概略図の部分である。

【図6】Profibus I/O通信プロトコルおよびASインタフェースI/O通信プロトコルを介して制御システム内で接続されているデバイスの文書化および構成を、FieldbusおよびHART通信プロトコルを介して制御システム内で接続されているデバイスと統合する図2の構成システムと関連する構成文書化階層概略図の部分である。

【図7】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図8】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図9】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図10】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図12】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図13】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図14】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図15】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文

書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図16】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図17】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図18】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図19】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図20】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図21】図1のプロセス制御システムのASインタフ

ェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図22】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

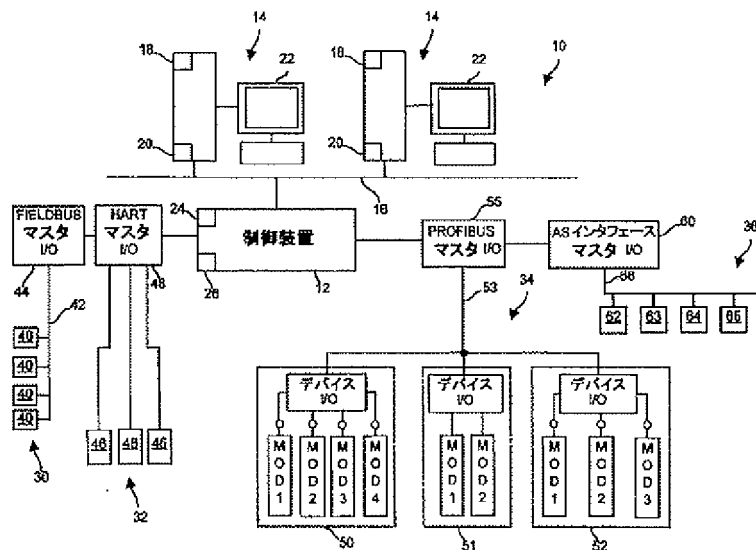
【図23】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図24】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

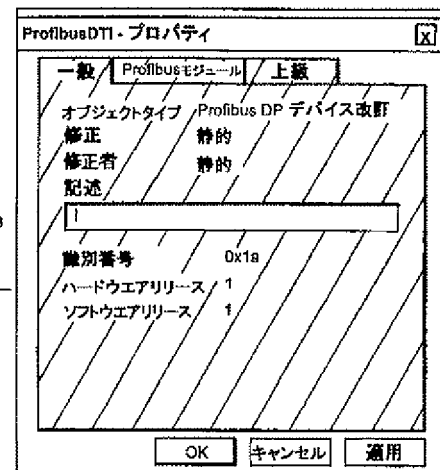
【図25】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

【図26】図1のプロセス制御システムのASインタフェースデバイスネットワーク要素の入力、構成および文書化を可能にするために、図2の構成システムによって使用されている例の画面の表示である。

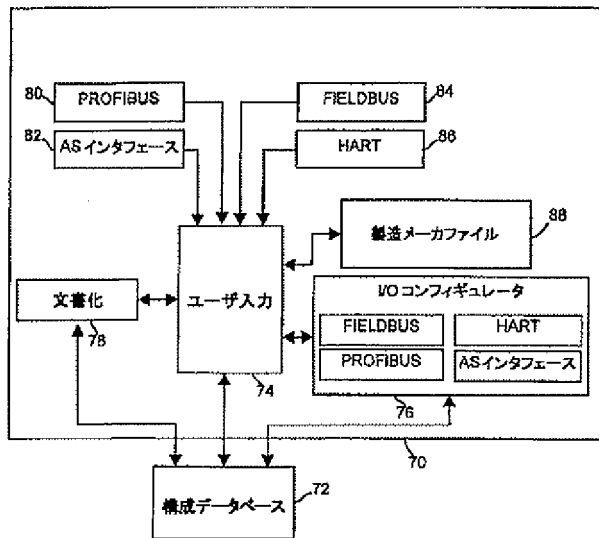
【図1】



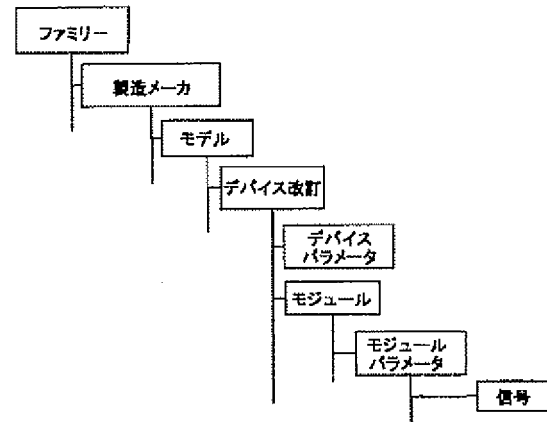
【図7】



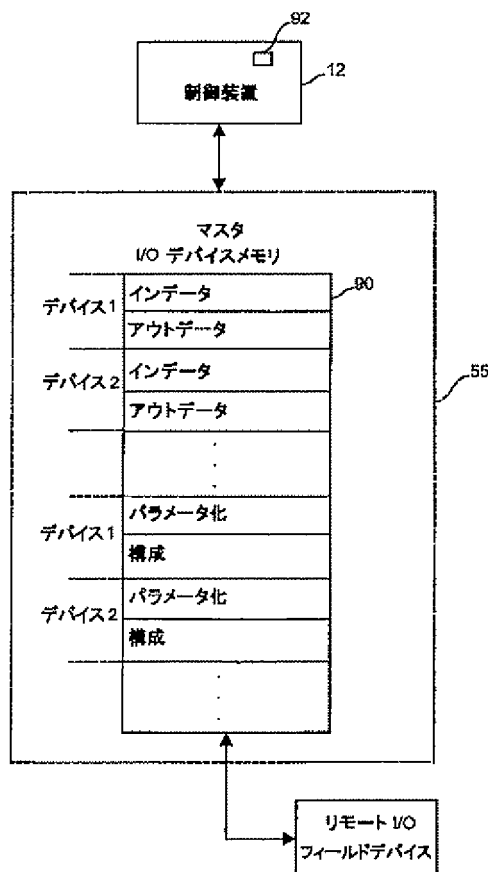
【図2】



【図3】



【図4】



【図5】

プロセス制御システム

ライブラリ
 Fieldbusデバイス
 製造メカ
 デバイス型
 デバイス改訂
 機能ブロック名
 実行時間
 インデックス

•
 •
 •

HARTデバイス
 製造メカ
 デバイス型
 デバイスI.D.
 記述
 診断プロパティ

•
 •
 •

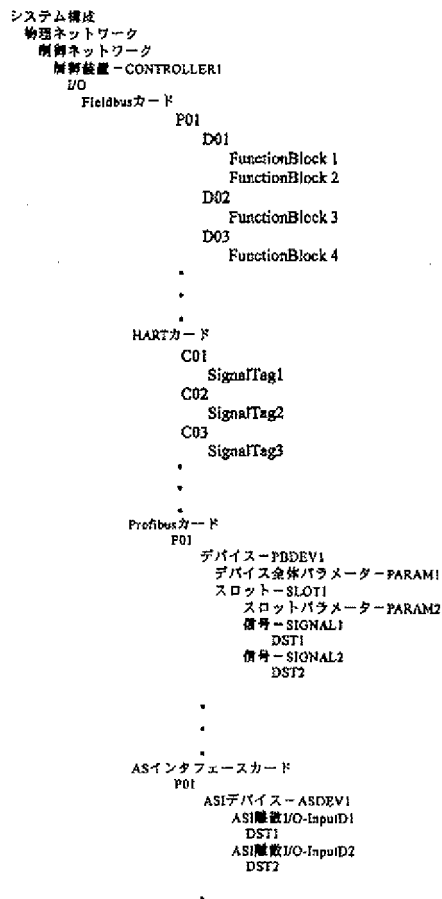
Profibusデバイス
 ファミリー=FAM1
 製造メカ=MAN1
 モデル=MODEL1
 デバイス改訂=REV1
 デバイス全体パラメータ=PARAM1
 モジュール=MODULE1
 パラメータ=PARAM2
 信号=SIGNAL1

•
 •
 •

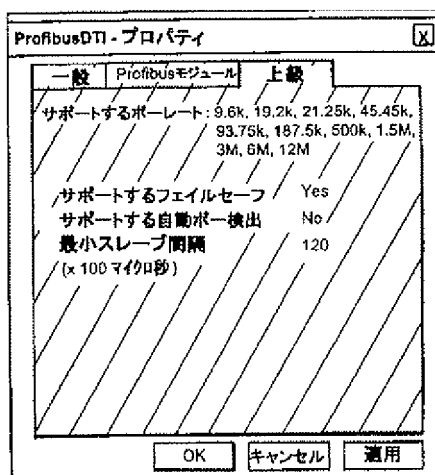
ASインタフェースデバイス
 ASIデバイス型=DeviceType1

•
 •
 •

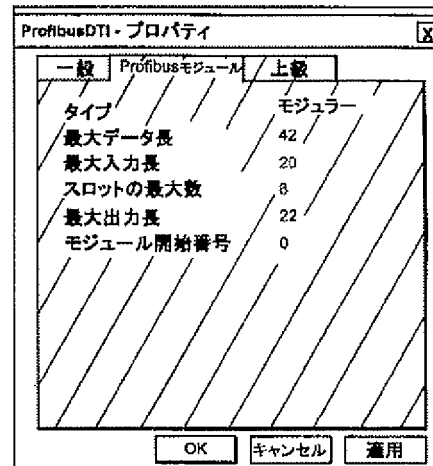
【図6】



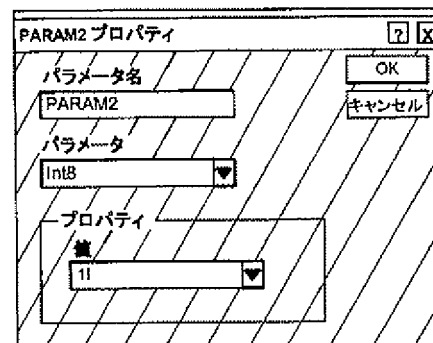
【図9】



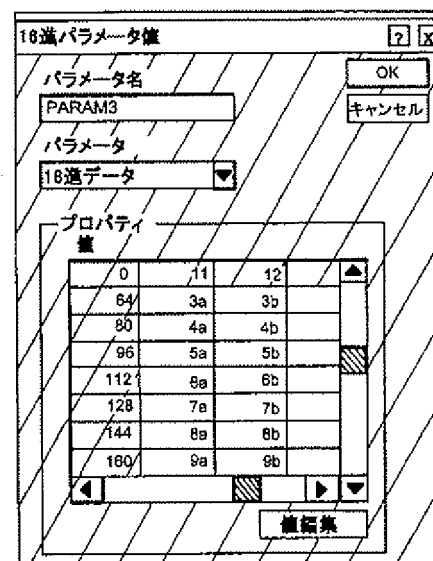
【図8】



【図10】



【図11】



【図12】

MODULE1 - プロパティ

オブジェクトタイプ Profibus DP モジュール

修正 静的

修正者 静的

記述

入力データ長 12

出力データ長 16

パラメータデータ長 24

識別子 0x01, 0x02

OK キャンセル

【図13】

SIGNAL1 - プロパティ

オブジェクトタイプ Profibus DP モジュール

修正 静的

修正者 静的

記述

データフォーマット

信号方向

データ型

バイトオフセット

開始ビット

ビット数

診断チャンネル

OK キャンセル

【図14】

Profibus DP モジュール ブラウザ

モジュール名	入力長	出力長
モジュール3	12	5
モジュール2	14	8
モジュール1	10	12

OK キャンセル

	許容最大値	使用	残余
入力バイト	64	16	48
出力バイト	64	16	48
トータルバイト	128	32	96
スロット数	16	4	12

【図15】

SLOT1 - プロパティ

オブジェクトタイプ Profibus DP スロット

修正 静的

修正者 静的

☐ 使用可

記述

モジュール数

入力データ長 12

出力データ長 16

パラメータデータ長 24

識別子 0x01, 0x02

OK キャンセル

【図16】

PO1 - プロパティ

オブジェクトタイプ Profibus DP ポート

修正 静的

修正者 静的

☐ 使用可

記述

ボーレート 9.6k

アドレス 21

OK Cancel キャンセル

【図18】

DeviceType1 - プロパティ

一般 プロファイル 入力 出力 パラメータ

プロファイル S-0.0

IO 構成 0x0 IN IN IN IN

識別子コード 0x0

OK キャンセル 適用 ヘルプ

【図17】

DeviceType1 - プロパティ

一般 プロファイル 入力 出力 パラメータ

オブジェクトタイプ AS デバイスタイプ

修正 火・9月 03 13:29:23 1998

修正者 Dove

記述

OK キャンセル 適用 ヘルプ

【図19】

DeviceType1 - プロパティ

一般 プロファイル 入力 出力 パラメータ

ラベル

11 ☒ []

12 ☐ []

13 ☐ []

14 ☐ []

OK キャンセル 適用 ヘルプ

【図21】

【図20】

DeviceType1 - プロパティ

一般 プロファイル 入力 出力 パラメータ

ラベル

01 ☒ []

02 ☒ []

03 ☐ []

04 ☐ []

OK キャンセル 適用 ヘルプ

DeviceType1 - プロパティ

一般 プロファイル 入力 出力 パラメータ

ラベル

P0 ☒ []

P1 ☐ []

P2 ☐ []

P3 ☒ []

OK キャンセル 適用 ヘルプ

【図23】

【図22】

P01 プロパティ

オブジェクトタイプ AS インターフェイスポート

修正 火・9月 03 13:29:23 1998

修正者 Dove

☐ 使用可

記述

コントローラのイベントのアクションは失敗?

☒ デバイスをリセット

☐ ボーリング確行

☐ 自動アドレスイネーブル

OK キャンセル

ASDEV1 Propeプロパティ

一般 パラメータ

オブジェクトタイプ AS デバイス

修正 火・9月 03 13:29:23 1998

修正者 Dove

☐ 使用可

名前

記述

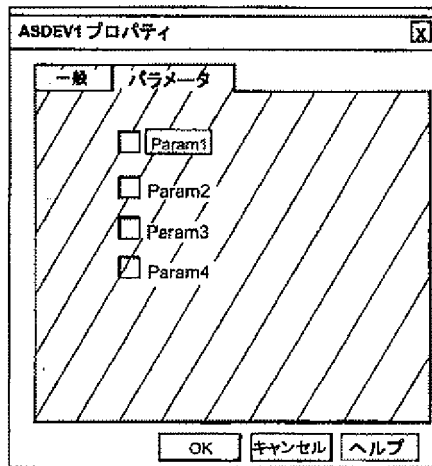
デバイスタイプ [] 新しいタイプ追加

プロファイル S-A.A

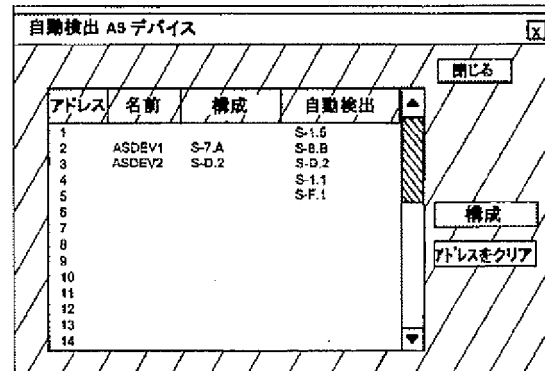
アドレス []

OK キャンセル ヘルプ

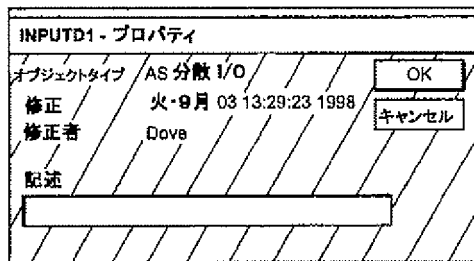
【図24】



【図25】



【図26】



【手続補正書】

【提出日】平成12年10月17日(2000.10.17)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】図面の簡単な説明

【補正方法】変更

【補正内容】

【図面の簡単な説明】

【図1】ローカルI/O、専用I/O、および遠隔I/Oデバイスネットワークに接続されている制御装置を有するプロセス制御システムのブロック図である。

【図2】プロセス制御システム内でローカルI/Oデバイスネットワーク、専用I/Oデバイスネットワークおよび遠隔I/Oデバイスネットワークを構成するために、ローカルI/Oデバイスネットワーク、専用I/Oデバイスネットワーク、および遠隔I/Oデバイスネットワークに関して、ユーザから情報を受け入れる構成システムのブロック図である。

【図3】図1のプロセス制御システムのプロセス制御構成システムで使用される信号オブジェクトデータベース

の一部のブロック図である。

【図4】図1のマスタI/Oデバイスで使用される共用メモリのブロック図である。

【図5】Profibus I/O通信プロトコルおよびASインタフェースI/O通信プロトコルを介して制御システム内で接続されているデバイスの文書化および構成を、FieldbusおよびHART通信プロトコルを介して制御システム内で接続されているデバイスと統合する図2の構成システムと関連する構成文書化階層概略図の部分である。

【図6】Profibus I/O通信プロトコルおよびASインタフェースI/O通信プロトコルを介して制御システム内で接続されているデバイスの文書化および構成を、FieldbusおよびHART通信プロトコルを介して制御システム内で接続されているデバイスと統合する図2の構成システムと関連する構成文書化階層概略図の部分である。

【図7】図1のプロセス制御システムのProfibusデバイスネットワーク要素の入力、構成、および文書化を可能にするために、図2の構成システムによって使

